Title
Iterative Algorithms for Distributed Optimization with Applications to Multi-Agent Estimation and Control

Author
Anhel Ferraz, Henrique

Publication Date
2019

Peer reviewed|Thesis/dissertation

University of California
Santa Barbara

Iterative Algorithms for Distributed Optimization with Applications to Multi-Agent Estimation and Control

A dissertation submitted in partial satisfaction
of the requirements for the degree

Doctor of Philosophy
in
Electrical and Computer Engineering

by

Henrique Anhel Ferraz

Committee in charge:

Professor João P. Hespanha, Chair
Professor Andrew R. Teel
Professor Francesco Bullo
Professor Jason R. Marden

December 2019

The Dissertation of Henrique Anhel Ferraz is approved.

Professor Andrew R. Teel

Professor Francesco Bullo

Professor Jason R. Marden

Professor João P. Hespanha, Committee Chair

November 2019

Iterative Algorithms for Distributed Optimization with Applications to Multi-Agent Estimation and Control

by

Henrique Anhel Ferraz

Acknowledgements

Throughout my graduate studies at UCSB I was privileged to be mentored by and receive the support of many special people. This dissertation could not have been completed without the generous spirit of professors, staff members, colleagues, friends, and family.

João, thank you for your guidance these past years. I am very grateful for your patience in letting me explore and develop my own research interests and ideas. Your sharp analytical skills, and pragmatic yet elegant approach for solving complex problems made navigating different topics and ideas a pleasant and smooth process.

Andy, thank you for always being available to answer quick math questions and for being the rock of CCDC. Francesco, I appreciate our interactions on network systems and graph theory. Your excitement and passion in teaching and research are truly inspiring. Jason, thank you for bringing new ideas and perspectives to CCDC, for all the career advice and conversations.

I am grateful for the camaraderie of my lab mates in HFH 5152 and 5156: Hari, Kyriakos, Masashi, Kunihisa, Hikaru, Michelle, Victor, Lucas, Pedro, Cuili, David, Justin, Matt, Murat, Sharad, Raphael, and Guosong. Thank you for your generosity, friendship, for the lunches together, social events, trips, discussions, and collaborations. I could not have asked for a better group to be part of.

CCDC has been an amazing community and I am fortunate to have shared my time with great fellow graduate students and postdocs: Nate, Ahmadreza, Keith, David, Philip, Matina, Sven, Brian, Bryce, Rahul, Saber, Dario, Xiaoming, and Herbert. A special thank you is reserved to Danny, Liz, and Jorge for their friendship and support during these years.

I am indebted to Val, Kallie, Stephanie, and Paul for making my academic life easier taking care of of the paperwork and so many of the processes that happen underneath the hood. Thank you also to my mentors at UFRJ Amit, Lizarralde, and Tiago at UERJ for helping me get to UCSB. I gratefully acknowledge the financial support for my research by CAPES under the grant BEX 1111-13-2, and by the National Science Foundation under the grants CNS-1329650 and EPCN-1608880. My collaborators at UCLA, Mani and Amr were vital to this work as well.

Outside CCDC, my dear friends David, Marc, Emily, Amanda, Simone, Jin, Celeste, Jaakko, Meri, Barbara, Mariela, Aristo, Geoff, and Arturo helped me have a good work-life balance and provided me comfort in times of trouble. A very special thank you is reserved for my little Brazilian community in Santa Barbara. Thank you Camilla, Osvaldo, Fernanda, Gabriel, Natasha, Alex, Pedro, Thati, Minhoca, Rafiel, Mari, Waltinho, and Walter for the many events and your endless support and friendship. Despite the distance, my friends from Papeizd, Palhinha, Pelada de 5ª, and UFRJ were fundamental during the past years being my virtual company in many moments and helping me manage my *saudade*.

Finally, and most importantly, I am indebted to my family in Rio and Belo Horizonte for their never-ending love and support. Clara, Louis, Beth, and João, you were indispensable for this dissertation to come to fruition.

Curriculum Vitæ

Henrique Anhel Ferraz

Education

2019	Ph.D. in Electrical and Computer Engineering (Expected), University of California, Santa Barbara
2017	M.S. in Electrical and Computer Engineering, University of California, Santa Barbara
2012	M.S. in Electrical Engineering, Federal University of Rio de Janeiro
2009	B.S. in Control and Automation Engineering, Federal University of Rio de Janeiro

Experience

2014 – 2019	Graduate Student Researcher, University of California, Santa Barbara
2013	Lecturer, *Analog Filter Design*, State University of Rio de Janeiro
2010 – 2012	Graduate Student Researcher, Federal University of Rio de Janeiro
2009	Undergraduate Research Assistant, Federal University of Rio de Janeiro

Publications

A. Alanwar, H. Ferraz, K. Hsieh, R. Thazhath, P. Martin, J. Hespanha, M. Srivastava. "D-SLATS: Distributed Simultaneous Localization and Time Synchronization," in *Proc. of the ACM MOBIHOC*, June 2017.

H. Ferraz, A. Alanwar, M. Srivastava and J. P. Hespanha, "Node localization based on distributed constrained optimization using Jacobi's method," in *IEEE 56th Annual Conference on Decision and Control (CDC)*, Melbourne, VIC, pp. 3380-3385, 2017.

H. Ferraz, J. Hespanha. "Iterative algorithms for distributed leader-follower model predictive control". in *IEEE 58th Annual Conference on Decision and Control (CDC)*, 2019. To appear

H. Ferraz, G. Yang, J. P. Hespanha, "Distributed algorithms for model predictive control based coordination of multi-agent systems", in preparation.

Abstract

Iterative Algorithms for Distributed Optimization with Applications to Multi-Agent Estimation and Control

by

Henrique Anhel Ferraz

Optimization is a prevalent tool in control and estimation. This work explores the theoretical and practical challenges in the design and analysis of distributed algorithms to solve optimization problems related to multi-agent systems.

We begin by considering a problem related to parameter estimation in sensor networks. We show that the maximum likelihood estimation formulation of several localization problems based on inter-sensor measurements reduces to the form of a common constrained optimization. We then design a distributed algorithm that utilizes only the most recent measurements and the estimates from the neighboring sensors, to iteratively compute the optimal solution. Our analysis shows that the solutions obtained from this algorithm converge locally to the maximum likelihood estimates, nevertheless simulations show that this convergence may occur globally. Furthermore, in experimental results using custom ultra-wideband radio frequency devices, this algorithm outperformed other distributed methods tested for a given localization problem.

Next, we consider a multi-agent coordination problem formulated as a finite horizon optimization of the type used in model predictive control. We present two distributed

and iterative algorithms, in which each agent is assigned a cost function, which it optimizes to compute its own control action. These cost functions depend on the states and the estimates of the control variables of the agents' neighbors, which are obtained through inter-agent communication. For the first algorithm, the agents are able to receive estimates from 2-hop neighbors, whereas the second algorithm utilizes only 1-hop neighbor information. For the first algorithm, our results show that the local solutions converge to the solution of the original model predictive control problem, regardless of how the algorithm is initialized. Because this convergence is asymptotic, we derive practical conditions for terminating the algorithm in a finite number of iterations, such that the closed-loop system achieves the desired coordination. For the second algorithm, due to more restrictive constraints, the convergence occurs to suboptimal solutions of the model predictive control problem. Nevertheless, simulations demonstrate that the optimality gap is small, and in some cases zero.

A key takeaway from these results is that in many problems of multi-agent systems, the communication between the agents can be leveraged to design distributed algorithms that match the quality of solutions that one would obtain from centralized approaches.

Contents

List of Figures

List of Symbols

- The set of complex numbers is denoted by $\mathbb{C}$, the set of real numbers is denoted by $\mathbb{R}$, and the set of (positive) integers is denoted by $(\mathbb{Z}_{>0})\mathbb{Z}$.
- The identity matrix in $\mathbb{R}^{n\times n}$ is denoted by I_n, whereas the zero matrix in $\mathbb{R}^{n\times m}$ is denoted by 0_{nm}, or simply by 0_n if $m = n$. The subscript will be dropped if the dimension is clear from the context.
- For a set $\mathcal{A}$, denote by $|\mathcal{A}|$ the cardinality of $\mathcal{A}$.
- The absolute value of a scalar r is denoted by $|r|$, the euclidean norm of a vector x by $\|x\|$, and the induced euclidean norm of a matrix A by $\|A\|$.
- A symmetric positive (semi)definite matrix is denoted by $(A \geqslant 0)$ $A > 0$.
- The maximum and minimum eigenvalue of a matrix A are denoted by $\bar{\lambda}(A)$ and $\underline{\lambda}(A)$, respectively.
- The ij element of a matrix A is denoted by $(A)_{ij}$.
- For twice continuously differentiable functions $f : \mathbb{R}^n \to \mathbb{R}^m$ and $g : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}$,

$\nabla_x f(x) \in R^{m\times n}$ denotes the Jacobian of f, $\nabla^2_{xx} f(x) \in \mathbb{R}^{n\times n}$ is the Hessian of f, and $\nabla^2_{xy} g(x, y) \in \mathbb{R}^{m\times n}$ denotes the second-order derivatives matrix of g.

- The Kronecker product of two matrices A and B is denoted by $A \otimes B$.

Chapter 1

Introduction

Optimization is a prevalent tool in control and estimation. This work explores the theoretical and practical challenges in the design and analysis of distributed algorithms to solve optimization problems found in multi-agent systems. When these problems involve the interaction of agents and distributed solutions are required, optimization becomes uniquely challenging. In this dissertation we studied distributed algorithms that emerge from two of such problems, the first one related to parameter estimation in sensor network, and the second related to control in multi-agent coordination problems. The two chapters that compose this dissertation are summarized as follows.

Distributed Localization in Sensor Networks (Chapter 2)

With the recent development of reliable, low-cost, and low-powered radio frequency and micro-electromechanical devices, several monitoring and control applications have benefited from the use of networks of wireless sensors and actuators. Examples include the inter-connection of appliances in homes, monitoring traffic in cities, environmental surveillance, precision agriculture, and tactical applications [10]. For many of these applications, determining the precise location of the devices is of utmost importance for the collected data to be meaningful [37].

However, in some cases the limitations in the environment and constraints on power and size make prohibitive the use of GPS to determine the location of the devices. Instead, the sensors must rely on inter-sensor measurements such as, angle-of-arrival, received signal strength, time-of-arrival, time-difference-of-arrival, and range to determine their positions [31]. In this chapter we consider the problem of determining the position of sensors from inter-sensor measurements.

Localization algorithms aim to combine available measurements and information to estimate the position of the sensors. The main challenges involve designing scalable methods that produce accurate estimates considering the efficient usage of processing power and bandwidth capacity. Based on how the measurements and information flow through the network and on how they are processed, the algorithms can be classified into two types: centralized or decentralized algorithms.

Centralized algorithms [45, 4] rely on a central device that has access to all the measurements between the sensors and single-handedly performs the computations needed to solve the localization problem. The solution obtained from centralized algorithms contains the position for all the nodes in the network. As the size of the network grows, these algorithms tend to demand more power, memory and communication to process the increasing number of measurements and variables. Additionally, centralized algorithms are vulnerable to single-point of failure and for these reasons are considered impractical for many applications.

Distributed algorithms offer an alternative to overcome these issues by exploiting the node's (sensor's) processing capability and the measurements that are available locally. The distributed localization therefore is comprised of a collection of smaller problems that can be solved at each node, or at a subset of the network. These methods tend to be scalable, robust, and adaptable to a variety of operation conditions [37].

However, technical and practical challenges arise from the distributed nature of this approach. First, because of power constraints, sensors are restricted to communicate with only a subset of the entire network, limiting the available information used to solve the localization problem. Additionally, executing these algorithms locally in general results in more power consumption from the devices reducing their life span. Finally, distributed algorithms analysis should provide theoretical guarantees for operations under a variety of scenarios such as node failures, transmission delays, and limited communication structure, while approaching quality of solutions that one would obtain from centralized approaches.

Contributions

We propose a distributed algorithm based on constrained optimization to localize nodes in a network, inspired by Jacobi's method as described in [5]. We show that the maximum likelihood estimation formulation of several localization problems based on inter-sensor measurements reduces to a constrained optimization with a specific structure. The constraints arise from the need to impose a coordinate system that avoids ambiguities arising from global rotations and translations.

The distributed algorithm iteratively computes the optimal solution using only the most recent measurements and estimates from the neighboring nodes. Additionally, at each node, only local variables need to be stored and transmitted, greatly reducing the complexity of the required local computations.

By regarding the iterative algorithm as a dynamical system and linearizing it around the optimal solution, we show that the algorithm converges to the maximum likelihood estimate, provided that it starts sufficiently close to it. While our stability results are local, simulations generally showed convergence to the optimal solution, regardless of how the algorithm is initialized. The proposed method was tested in hardware using custom ultra-wideband radio frequency devices outperforming other distributed methods found in the literature.

Related work

Both centralized and distributed approaches to localization have been extensively investigated by the wireless sensor network community. Evaluation of the accuracy and the computation of performance bounds for any unbiased location estimators are addressed in [38], where the authors propose a framework for comparison of different algorithms. A centralized convex optimization scheme is proposed in [15], where the communication network is modeled as a set of geometric constraints. Distributed localization techniques can be based on multidimensional scaling and coordinate alignment techniques [26], multilateration [41, 33], and graph-theoretical methods [3]. In [36] the authors propose a hop-by-hop connectivity-based algorithm using trilateration. For range-based and range-free localization problems, [42] presents a convex formulation obtained by relaxation techniques that is solved with a sequential greedy optimization algorithm. Our approach builds upon [5], where it is proposed an algorithm that combines the maximum likelihood estimate and the Jacobi method for localization with relative position measurements.

From an algorithmic point of view, the methods presented in this chapter and in the chapter that follows can be classified to be distributed optimization methods. In this context, the seminal work by Tsitsiklis [46] sets the foundations for the analysis of distributed optimization algorithms. Since then, many works have focused on designing discrete-time algorithms to find the solution of optimization problems, where the cost function is a sum of convex functions [39, 27, 34, 7]. The increased interest in the design and analysis of distributed optimization methods comes from their applicability in several types of problems, such as localization and coordination as discussed in this dissertation.

For problems with constraints, a distributed primal-dual subgradient method was proposed in [48], where the global constraint set is the intersection of local constraints set. A similar problem setup was studied in [35] for networks with time-varying connectivity, for which a consensus-based distributed algorithm was presented. More recently, algorithms that deal with distributed continuous-time strategies were investigated in [21] and in [29], where a more sophisticated combination of a local continuous-time and discrete-time dynamics for communication with the neighbors was proposed.

Distributed Coordination for Multi-Agent Systems (Chapter 3)

Large engineering systems are in general composed of multiple subsystems that interact with each other sharing and exchanging resources of energy, information, and material. While some systems have natural and immutable constraints on their coupling, we consider the case where we design the coupling between the parts taking into account trade-offs in performance and communication costs.

In the past, control of multiple agents has received enormous attention due to the benefits obtained when a single complex system task is replaced by the coordinated actions of multiple and simpler subsystems. Multi-agent coordination has applications in vehicle platooning [16], flight formation [8], surveillance operations [13], wireless sensor network [37], and in many other problems.

We consider a multi-agent coordination problem where the objective is to design control inputs for a collection of agents to follow a single leader. The systems have identical dynamics, possibly unstable, and only a fraction of the followers communicate with the leader. We formulate the problem as a finite horizon optimization of the type used in model predictive control [40], where the cost is given by the sum of local cost functions.

Similar to localization, there are two distinct approaches adopted for controlling multiple agents. The centralized approach is based on the existence of a central agent (that could be part of the coordination problem or not) that collects all the relevant data and determines the actions for each agent such that the overall system accomplishes the desired goal. Because of their structure, these methods are considered less flexible although good performance results are reported in applications [8]. On the other hand, distributed approach does not require a centralize agent but rather relies on the coordinated solution of local problems. This approach offers many benefits from the scalability of the algorithm to robustness and adaptability.

Model predictive control (MPC) provides a reliable framework for a range of applications, utilizing a dynamical model of the system and optimizing over a forecast window. From a control design perspective, the types of solutions proposed in this chapter fit into a distributed model predictive control (DMPC) framework. The key difference between distributed and other types of approaches that deal with systems composed of multiple parts that interact, such as decentralized model predictive control, lies in the fact that in DMPC some communication may be established between the different subsystems to achieve stability or to improve performance.

Contributions

The main contributions of this chapter are two iterative and distributed algorithms, in which each agent is assigned a cost function that it optimizes to compute its own control action. These algorithms have different cost functions, although both costs depend on the state and the estimates of the control variables of the agents neighbors, which are obtained through inter-agent communication. In the first algorithm, the agents are required to get these estimates from 2-hop neighbors, whereas the second algorithm requires less communication, only needing estimates from 1-hop neighbors.

For the first algorithm, we show that the local solutions converge to the centralized optimal solution of the original MPC problem, regardless of how the algorithm is initialized. Since this convergence occurs asymptotically, we derive practical conditions to terminate the algorithm in a finite number of iterations, such that when the resulting control estimates are applied, the desired coordination is achieved. For the second algorithm, due to more restrictive constraints in the range of communication, the estimates converge generally to sub-optimal solutions. Nevertheless, simulations show that the optimality gap is small, and in some special cases zero. The proposed methods are evaluated in simulations for randomly generated graphs and for agents with unstable dynamics.

Related work

The study developed in this chapter can be seen through the lens of two related, but distinct subjects. From a purely algorithmic point of view, our approach builds on the topic of distributed optimization. For a discussion of the relevant literature and algorithms we refer the reader to the related work section for Chapter 2.

From a control design perspective our approach can be seen as a DMPC problem. A recent overview of distributed approaches for MPC can be found in the survey [11]. Our work lies at the intersection of non-cooperative DMPC, where each local controller optimizes a local cost function; and cooperative DMPC, in which local controllers optimize a common global cost function, as defined in [40]. This is because we propose to optimize a global cost function by designing appropriate local cost functions and solving these optimizations locally and iteratively.

The methods presented in this chapter are similar to the ones found in [43] and [47], where the solutions for a centralized problem are approximated by a succession of iterations. However, in our approach we do not require an upper bound on the number of unstable modes of the system dynamics and we explicitly address in the problem formulation the graph structure spanned by the communication between the agents, deriving sufficient conditions for the closed-loop stability to hold.

Similar types of multi-agent coordination problems using DMPC approaches have been study in the literature, but many of these works rely on fully connected network assumptions[20, 32, 30], which can be overly restrictive in some applications. For a vehicle formation control problem, [16] proposes a finite receding horizon control and establishes stability results, but provides no guarantees that the local performance matches the performance resulting from a centralized implementation. A distributed output-feedback model predictive control that combines simultaneous state estimation with control computation has been studied in [12] providing practical consensus results. While most of these results guarantee synchronization or the convergence to a predetermined agent formation, the analysis of the designed DMPC algorithm performance in comparison to a centralized implementation has been overlooked.

Chapter 2

Distributed Localization in Sensor Networks

Parts of this chapter come from [17] and [2].

In this chapter we consider the problem of localizing devices in sensor networks. We begin by generalizing several maximum likelihood estimation problems based on inter-sensor measurements to a common constrained optimization form. Then, we leverage the inter-sensor communication to design an iterative algorithm that utilizes the measurements and the most recent estimates from the neighbors to compute this optimal solution. Theoretical results demonstrate that the solutions produced by the algorithm converge locally to the maximum likehood estimate. However, simulations show that when the graphs satisfy a certain structural condition, this convergence is in general global. Finally, experimental tests using radio frequency devices are presented for mobile and static sensors, for a combined localization and time synchronization type of problem. The results show that the method achieves 30 cm localization error and 3 micro-seconds time synchronization error, outperforming two other distributed methods tested.

Structure This chapter is organized as follows. In Section 2.1, we formulate an optimization problem that generalizes several problems related to sensor (node) localization and time synchronization. Section 2.2 introduces a distributed algorithm to solve the optimizations and presents conditions for local asymptotic convergence. A case study for a range-based localization is presented in Section 2.3 and numerical simulations illustrate the proposed algorithm. In Section 2.4 we evaluate the proposed method in an experimental setup using ultra-wideband radio devices for a combined localization and time synchronization type of problem. We conclude the chapter in Section 2.5 summarizing the theoretical, numerical and experimental results obtained.

2.1 Problem Formulation

Several problems related to the localization of multi-agent systems can be reduced to optimizations of the following form:

$$\min_{x} \quad \sum_{i\in\mathcal{N}} f_i(x_i) + \sum_{i\in\mathcal{N}} \sum_{j\in\mathcal{N}_i} f_{ij}(x_i, x_j), \tag{2.1a}$$

$$\text{subject to} \quad h_i(x_i) = 0, \quad \forall i \in \mathcal{N}, \tag{2.1b}$$

where $x \coloneqq (x_1, x_2, \dots, x_N) \in \mathbb{R}^{n_1} \times \mathbb{R}^{n_2} \times \cdots \times \mathbb{R}^{n_N}$ are the optimization variables, $\mathcal{N} \coloneqq \{1, 2, \dots, N\}$, $\mathcal{N}_i$ is a subset of $\mathcal{N} \backslash \{i\}$ containing the neighbors of node i, and the functions $f_i : \mathbb{R}^{n_i} \to \mathbb{R}$, $f_{ij} : \mathbb{R}^{n_i} \times \mathbb{R}^{n_j} \to \mathbb{R}$, $h_i : \mathbb{R}^{n_i} \times \mathbb{R}^{n_j} \to \mathbb{R}^{m_i}$, $i \in \mathcal{N}, j \in \mathcal{N}_i$ are all twice continuously differentiable.

We consider here four problems, where we want to localize in space a set $\mathcal{N} \coloneqq \{1, 2, \dots, N\}$ of N nodes based on relative measurements of each node $i \in \mathcal{N}$ with respect to its neighbors $\mathcal{N}_i \subset \mathcal{N} \backslash \{i\}$.

2.1.1 Relative Position Measurements

In this scenario, each variable $p_i \in \mathbb{R}^d$, $d = \{2, 3\}$, $i \in \mathcal{N}$ denotes the position of node i in a global coordinate system and the node i has access to noisy measurements $z_{ij} \in \mathbb{R}^d$ of the relative position $p_j - p_i$ of each neighboring node $j \in \mathcal{N}_i$. Specifically,

$$z_{ij} = p_j - p_i + w_{ij}, \qquad \forall i \in \mathcal{N},\ j \in \mathcal{N}_i,$$

where the w_{ij} denote independent zero-mean Gaussian noise with co-variance matrix $\Sigma_{ij} > 0$. For this problem, the symmetric of the log-likelihood of the measurements $\{z_{ij} \in \mathbb{R}^d : i \in \mathcal{N},\ j \in \mathcal{N}_i\}$ is given by

$$\frac{1}{2} \sum_{i \in \mathcal{N}} \sum_{j \in \mathcal{N}_j} (p_i - p_j + z_{ij})^\top \Sigma_{ij}^{-1} (p_i - p_j + z_{ij}).$$

Just with relative measurements it is possible to only reconstruct the positions p_i up to a global translation. To avoid this ambiguity one can force the position of one "reference" node (say node $i = 1$) to be the origin of the coordinate system, which corresponds to the constraint

$$p_1 = 0_{n1}.$$

The computation of maximum likelihood estimates for the p_i thus amounts to solving an optimization of the form (2.1) with

$$x_i \coloneqq p_i, \qquad f_i(x_i) \coloneqq 0, \qquad f_{ij}(x_i, x_j) \coloneqq \frac{1}{2}(p_i - p_j + z_{ij})^\top \Sigma_{ij}^{-1}(p_i - p_j + z_{ij}),$$
$$h_1(x_1) \coloneqq p_1, \qquad h_i(x_i) \coloneqq 0, \quad \forall i \in \{2, 3, \dots, N\}.$$

When the neighborhoods $\mathcal{N}_i$ induce a graph in which there is a path from the reference node 1 to every other node, this optimization is a strictly convex quadratic program [5].

In this formulation and the ones that follow, we ignore any prior information about the positions p_i. When prior distributions for these variables are available, this information could be incorporated into the optimization through the functions $f_i(x_i)$.

For this problem, we have that

$$\nabla_{x_i} f_{ij}(x_i, x_j) = (p_i - p_j + z_{ij})^\top \Sigma_{ij}^{-1}, \qquad \forall i, j, \tag{2.2a}$$
$$\nabla_{x_j} f_{ij}(x_i, x_j) = -(p_i - p_j + z_{ij})^\top \Sigma_{ij}^{-1}, \qquad \forall i, j, \tag{2.2b}$$

$$\nabla^2_{x_i x_i} f_{ij}(x_i, x_j) = \nabla^2_{x_j x_j} f_{ij}(x_i, x_j) = \Sigma_{ij}^{-1}, \qquad \forall i, j, \tag{2.2c}$$

$$\nabla^2_{x_i x_j} f_{ij}(x_i, x_j) = \nabla^2_{x_j x_i} f_{ij}(x_i, x_j) = -\Sigma_{ij}^{-1}, \qquad \forall i \neq j, \tag{2.2d}$$

$$\nabla_{x_1} h_1(x_1) = I_n, \tag{2.2e}$$

$$\nabla_{x_i} h_i(x_i) = 0, \qquad \forall i > 1, \tag{2.2f}$$

$$\nabla^2_{x_i x_i} h_i(x_i) = 0, \qquad \forall i. \tag{2.2g}$$

2.1.2 Range Measurements

This scenario is analogous to the previous one, but now the node i has access to noisy measurements $z_{ij} \in \mathbb{R}$ of its distance $\|p_j - p_i\|$ to each of its neighboring nodes $j \in \mathcal{N}_i$. Specifically,

$$z_{ij} = \|p_j - p_i\| + w_{ij}, \qquad \forall i \in \mathcal{N},\ j \in \mathcal{N}_i,$$

where the w_{ij} denote independent zero-mean Gaussian noise with variance $\sigma_{ij} > 0$. For this problem, the symmetric of the log-likelihood of the measurements $\{z_{ij} \in \mathbb{R} : i \in \mathcal{N},\ j \in \mathcal{N}_i\}$ is given by

$$\frac{1}{2} \sum_{i \in \mathcal{N}} \sum_{j \in \mathcal{N}_j} \frac{(\|p_j - p_i\| - z_{ij})^2}{\sigma_{ij}^2}.$$

Just with relative measurements, it is only possible to reconstruct the positions $p_i \in \mathbb{R}^d$ up to a global translation and rotation. To avoid this ambiguity one can force the position of one "reference" node (say node 1) to be the origin of the coordinate system and, for $d = 3$, use two other nodes, say nodes 2 and 3, to define the orientation of the coordinate system. Specifically, forcing the 1st axis of the coordinate system to be aligned with the vector from p_1 to p_2 and the second axis to lie in the plane defined by the first 3 nodes (assumed not to be co-linear), corresponds to the constraints

$$p_1 = 0, \qquad e_2^\top p_2 = e_3^\top p_2 = 0, \qquad e_3^\top p_3 = 0,$$

where $e_i \in \mathbb{R}^3$ denotes the ith vector of the canonical basis of $\mathbb{R}^3$. The computation of maximum likelihood estimate for the p_i thus amounts to solving an optimization of the form (2.1) with

$$x_i \coloneqq p_i, \qquad f_i(x_i) \coloneqq 0, \qquad f_{ij}(x_i, x_j) \coloneqq \frac{1}{2}\frac{(\|p_j - p_i\| - z_{ij})^2}{\sigma_{ij}^2},$$

$$h_1(x_1) \coloneqq p_1, \qquad h_2(x_2) \coloneqq \begin{bmatrix} e_2^\top \\ e_3^\top \end{bmatrix} p_2, \qquad h_3(x_3) \coloneqq e_3^\top p_3, \quad h_i(x_i) \coloneqq 0, \quad \forall i \in \{3, \ldots, N\}.$$

For the case where $d = 2$, only two nodes are needed to define the coordinate system.

When the neighborhoods $\mathcal{N}_i$ induce a framework that is rigid and the measurements are noiseless, this optimization has an isolated global minima at the true positions of the nodes [4].

For this problem, we have that

$$\begin{aligned}
\nabla_{x_i} f_{ij}(x_i, x_j) &= -\frac{1}{2}\frac{\|p_j - p_i\| - z_{ij}}{\sigma_{ij}^2 \|p_j - p_i\|}(p_j - p_i)^\top \\
&= -\frac{1}{2}\Big(\frac{1}{\sigma_{ij}^2} - \frac{z_{ij}}{\sigma_{ij}^2\|p_j - p_i\|}\Big)(p_j - p_i)^\top, && \forall i, j, \\
\nabla_{x_j} f_{ij}(x_i, x_j) &= \frac{1}{2}\frac{\|p_j - p_i\| - z_{ij}}{\sigma_{ij}^2 \|p_j - p_i\|}(p_j - p_i)^\top \\
&= \frac{1}{2}\Big(\frac{1}{\sigma_{ij}^2} - \frac{z_{ij}}{\sigma_{ij}^2\|p_j - p_i\|}\Big)(p_j - p_i)^\top, && \forall i, j, \\
\nabla^2_{x_i x_i} f_{ij}(x_i, x_j) &= \nabla^2_{x_j x_j} f_{ij}(x_i, x_j) \\
&= \frac{1}{2}\Big(\frac{1}{\sigma_{ij}^2} - \frac{z_{ij}}{\sigma_{ij}^2\|p_j - p_i\|}\Big) I_3 + \frac{1}{4}\frac{z_{ij}}{\sigma_{ij}^2\|p_j - p_i\|^3}(p_j - p_i)(p_j - p_i)^\top \\
&= \frac{1}{2}\frac{\|p_j - p_i\| - z_{ij}}{\sigma_{ij}^2 \|p_j - p_i\|} I_3 + \frac{1}{4}\frac{z_{ij}}{\sigma_{ij}^2\|p_j - p_i\|^3}(p_j - p_i)(p_j - p_i)^\top, && \forall i, j, \\
\nabla^2_{x_i x_j} f_{ij}(x_i, x_j) &= \nabla^2_{x_j x_i} f_{ij}(x_i, x_j) \\
&= -\frac{1}{2}\frac{\|p_j - p_i\| - z_{ij}}{\sigma_{ij}^2 \|p_j - p_i\|} I_3 - \frac{1}{4}\frac{z_{ij}}{\sigma_{ij}^2\|p_j - p_i\|^3}(p_j - p_i)(p_j - p_i)^\top, && \forall i \neq j,
\end{aligned}$$

$$\nabla_{x_1} h_1(x_1) = I_3, \qquad \nabla_{x_2} h_2(x_2) = \begin{bmatrix} e_2^\top \\ e_3^\top \end{bmatrix}, \qquad \nabla_{x_3} h_3(x_3) = e_3^\top,$$

$$\nabla_{x_i} h_i(x_i) = 0, \quad \forall i \in \{3, \ldots, N\}, \qquad \nabla^2_{x_i x_i} h_i(x_i) = 0, \quad \forall i.$$

In the absence of noise (i.e., $w_{ij} = 0$) and for values of p_i, p_j compatible with the measured distance z_{ij} (i.e., $\|p_j - p_i\| = z_{ij}$), the Hessian formulas simplify to

$$\nabla^2_{x_i x_i} f_{ij}(x_i, x_j) = \nabla^2_{x_j x_j} f_{ij}(x_i, x_j) = \frac{1}{4\sigma_{ij}^2} s_{ij} s_{ij}^\top, \qquad \forall i, j,$$

where $s_{ij} := \frac{1}{\|p_j - p_i\|}(p_j - p_i)$ denotes the unit vector pointing from p_i to p_j.

2.1.3 Pseudo-range Measurements

In this scenario, each node $i \in \mathcal{N}$ broadcasts a wireless message to its neighbors $j \in \mathcal{N}_i$ with the value t_i of its local clock at the transmission time and the neighbors record the times of arrival t_{ij} of this message in their local clocks. The clock of each node i has an unknown offset τ_i with respect to the "global" time reference and therefore the actual time at which the message was transmitted by node i is given by

$$t_i - \tau_i + w_i$$

and the time at which the message was received by node j is given by

$$t_{ij} - \tau_j + w_{ij},$$

where w_i and w_{ij} denote time measurement errors. Assuming that messages propagate at a velocity c, we have that

$$t_{ij} - \tau_j + w_{ij} = t_i - \tau_i + w_i + \frac{\|p_j - p_i\|}{c},$$

which can be re-written as

$$t_{ij} - t_i = \tau_j - \tau_i + \frac{\|p_j - p_i\|}{c} + \bar{w}_{ij},$$

where $\bar{w}_{ij} \coloneqq w_i - w_{ij} \in \mathbb{R}$. Assuming that the $\bar{w}_{ij}$ are independent zero-mean Gaussian random variables with variances $\sigma_{ij} > 0$, the symmetric of the log-likelihood of the measurements $\{z_{ij} \in \mathbb{R} : i \in \mathcal{N},\ j \in \mathcal{N}_i\}$ is given by

$$\frac{1}{2} \sum_{i \in \mathcal{N}} \sum_{j \in \mathcal{N}_j} \frac{(\tau_j - \tau_i + \frac{\|p_j - p_i\|}{c} - t_{ij} + t_i)^2}{\sigma_{ij}^2}.$$

In this problem, we have ambiguity with respect to a global rotation and translation, as the range measurements case in Section 2.1.2, but also with respect to a shift of the time reference, which can be resolved by forcing the clock of node 1 to determine the global time t, so that the offsets for every clock will be with respect to the clock at node 1. This corresponds to the following constraints:

$$p_1 = 0, \qquad e_2^\top p_2 = e_3^\top p_2 = 0, \qquad e_3^\top p_3 = 0, \qquad \tau_1 = 0.$$

The computation of maximum likelihood estimates for the positions p_i and the clock offsets τ_i thus amounts to solving an optimization of the form (2.1) with

$$x_i \coloneqq \begin{bmatrix} p_i \\ \tau_i \end{bmatrix}, \qquad f_i(x_i) \coloneqq 0, \qquad f_{ij}(x_i, x_j) \coloneqq \frac{1}{2}\frac{(\tau_j - \tau_i + \frac{\|p_j - p_i\|}{c} - t_{ij} + t_i)^2}{\sigma_{ij}^2},$$

$$h_1(x_1) \coloneqq \begin{bmatrix} p_1 \\ \tau_1 \end{bmatrix}, \qquad h_2(x_2) \coloneqq \begin{bmatrix} e_2^\top \\ e_3^\top \end{bmatrix} p_2, \qquad h_3(x_3) \coloneqq e_3^\top p_3, \qquad h_i(x_i) \coloneqq 0, \quad \forall i > 3.$$

For this problem, we have that $\forall i, j$

$$\begin{aligned}
&\nabla_{x_i} f_{ij}(x_i, x_j) \\
&\quad = \begin{bmatrix} -\frac{1}{2}\frac{\tau_j - \tau_i + \frac{\|p_j - p_i\|}{c} - t_{ij} + t_i}{\sigma_{ij}^2}\frac{1}{c\|p_j - p_i\|}(p_j - p_i)^\top & -\frac{\tau_j - \tau_i + \frac{\|p_j - p_i\|}{c} - t_{ij} + t_i}{\sigma_{ij}^2} \end{bmatrix} \\
&\quad = \begin{bmatrix} -\frac{1}{2c^2\sigma_{ij}^2}(p_j - p_i)^\top - \frac{\tau_j - \tau_i - t_{ij} + t_i}{2c\sigma_{ij}^2\|p_j - p_i\|}(p_j - p_i)^\top & -\frac{\tau_j - \tau_i - t_{ij} + t_i}{\sigma_{ij}^2} - \frac{\|p_j - p_i\|}{c\sigma_{ij}^2} \end{bmatrix},
\end{aligned}$$

$$\begin{aligned}
&\nabla_{x_j} f_{ij}(x_i, x_j) \\
&\quad = \begin{bmatrix} \frac{1}{2c^2\sigma_{ij}^2}(p_j - p_i)^\top + \frac{\tau_j - \tau_i - t_{ij} + t_i}{2c\sigma_{ij}^2\|p_j - p_i\|}(p_j - p_i)^\top & \frac{\tau_j - \tau_i - t_{ij} + t_i}{\sigma_{ij}^2} + \frac{\|p_j - p_i\|}{c\sigma_{ij}^2} \end{bmatrix},
\end{aligned}$$

$$\begin{aligned}
&\nabla^2_{x_i x_i} f_{ij}(x_i, x_j) = \nabla^2_{x_j x_j} f_{ij}(x_i, x_j) \\
&\quad = \begin{bmatrix} \frac{1}{2c^2\sigma_{ij}^2} I_3 + \frac{\tau_j - \tau_i - t_{ij} + t_i}{2c\sigma_{ij}^2\|p_j - p_i\|} I_3 - \frac{\tau_j - \tau_i - t_{ij} + t_i}{4c\sigma_{ij}^2\|p_j - p_i\|^3}(p_j - p_i)(p_j - p_i)^\top & \frac{1}{2c\sigma_{ij}^2\|p_j - p_i\|}(p_j - p_i) \\ \frac{1}{2c\sigma_{ij}^2\|p_j - p_i\|}(p_j - p_i)^\top & \frac{1}{\sigma_{ij}^2} \end{bmatrix} \\
&\quad = \begin{bmatrix} \frac{\tau_j - \tau_i + \frac{\|p_j - p_i\|}{c} - t_{ij} + t_i}{2c\sigma_{ij}^2\|p_j - p_i\|}\left(I_3 - \frac{1}{2\|p_j - p_i\|^2}(p_j - p_i)(p_j - p_i)^\top\right) & 0 \\ 0 & 0 \end{bmatrix}
\end{aligned}$$

$$+\frac{1}{\sigma_{ij}^2}\begin{bmatrix}\frac{1}{2c\|p_j-p_i\|}(p_j-p_i)\\ 1\end{bmatrix}\begin{bmatrix}\frac{1}{2c\|p_j-p_i\|}(p_j-p_i)\\ 1\end{bmatrix}^\top$$

$$\nabla^2_{x_ix_j}f_{ij}(x_i,x_j)=\nabla^2_{x_jx_i}f_{ij}(x_i,x_j)$$
$$=\begin{bmatrix}-\frac{\tau_j-\tau_i+\frac{\|p_j-p_i\|}{c}-t_{ij}+t_i}{2c\sigma_{ij}^2\|p_j-p_i\|}\left(I_3-\frac{1}{2\|p_j-p_i\|^2}(p_j-p_i)(p_j-p_i)^\top\right) & 0\\ 0 & 0\end{bmatrix}$$
$$-\frac{1}{\sigma_{ij}^2}\begin{bmatrix}\frac{1}{2c\|p_j-p_i\|}(p_j-p_i)\\ 1\end{bmatrix}\begin{bmatrix}\frac{1}{2c\|p_j-p_i\|}(p_j-p_i)\\ 1\end{bmatrix}^\top, \quad i\neq j$$

$$\nabla_{x_1}h_1(x_1)=I_4, \qquad \nabla_{x_2}h_2(x_2)=\begin{bmatrix}e_2^\top\\ e_3^\top\end{bmatrix}, \qquad \nabla_{x_3}h_3(x_3)=e_3^\top,$$

$$\nabla_{x_i}h_i(x_i)=0, \quad \forall i\in\{3,\dots,N\}, \qquad \nabla^2_{x_ix_i}h_i(x_i)=0.$$

In the absence of noise (i.e., $\bar{w}_{ij}=0$) and for values of p_i,p_j,τ_i,τ_j compatible with the measurements t_i,t_{ij} (i.e., $t_{ij}-t_i=\tau_j-\tau_i+\frac{\|p_j-p_i\|}{c}$), the Hessian formulas simplify to

$$\nabla^2_{x_ix_i}f_{ij}(x_i,x_j)=\nabla^2_{x_jx_j}f_{ij}(x_i,x_j)=\frac{1}{\sigma_{ij}^2}v_{ij}v_{ij}^\top, \qquad \forall i,j,$$

where

$$v_{ij}\coloneqq\begin{bmatrix}\frac{1}{2c}s_{ij}\\ 1\end{bmatrix},$$

and $s_{ij}\coloneqq\frac{1}{\|p_j-p_i\|}(p_j-p_i)$ denotes the unit vector pointing from p_i to p_j.

2.1.4 Pseudo-range Measurements with Clock Drift and Biases

This scenario is similar to the previous one, but now we consider clock drifts and biases in the time measurements. In this case, the transmission time measurement t_i for a message sent at time t (at a global reference clock) by node i is given by

$$t_i = \phi_i t + \tau_i + w_i \quad \Leftrightarrow \quad t = \frac{t_i - \tau_i - w_i}{\phi_i}$$

and the reception time measurement t_{ij} for a message received at time t (at a global reference clock) by node j is given by

$$t_{ij} = \phi_j t + \tau_j + w_{ij} \quad \Leftrightarrow \quad t = \frac{t_{ij} - \tau_j - w_{ij}}{\phi_j},$$

where ϕ_i is the clock drift of node i, τ_i a bias in the transmission-time measurement for node i, τ_j a bias in the reception-time measurement for node j, and w_i, w_{ij} zero-mean noise.

Assuming that messages propagate at a velocity c, we have that

$$\frac{t_{ij} - \tau_j - w_{ij}}{\phi_j} = \frac{t_i - \tau_i - w_i}{\phi_i} + \frac{\|p_j - p_i\|}{c},$$

which can be re-written as

$$\phi_j^{-1}t_{ij} - \phi_i^{-1}t_i = \bar{\tau}_j - \bar{\tau}_i + \frac{\|p_j - p_i\|}{c} + \bar{w}_{ij},$$

where $\bar{w}_{ij} \coloneqq \phi_i^{-1}w_i - \phi_j^{-1}w_{ij} \in \mathbb{R}$, $\bar{\tau}_j = \phi_j^{-1}\tau_j$. Assuming that the $\bar{w}_{ij}$ are independent zero-mean Gaussian random variables with variances $\sigma_{ij} > 0$, the symmetric of the log-likelihood of the measurements $\{z_{ij} \in \mathbb{R} : i \in \mathcal{N},\ j \in \mathcal{N}_i\}$ is given by

$$\frac{1}{2}\sum_{i\in\mathcal{N}}\sum_{j\in\mathcal{N}_j}\frac{(\bar{\tau}_j - \bar{\tau}_i + \frac{\|p_j-p_i\|}{c} - \phi_j^{-1}t_{ij} + \phi_i^{-1}t_i)^2}{\sigma_{ij}^2}.$$

The computation of maximum likelihood estimates for the p_i, τ_i, and ϕ_i, thus amounts to solving an optimization of the form (2.1) with

$$x_i \coloneqq \begin{bmatrix} p_i \\ \bar{\tau}_i \\ \phi_i^{-1} \end{bmatrix}, \qquad f_i(x_i) \coloneqq 0, \qquad f_{ij}(x_i, x_j) \coloneqq \frac{1}{2}\frac{(\bar{\tau}_j - \bar{\tau}_i + \frac{\|p_j-p_i\|}{c} - \phi_j^{-1}t_{ij} + \phi_i^{-1}t_i)^2}{\sigma_{ij}^2},$$

$$h_1(x_1) \coloneqq \begin{bmatrix} p_1 \\ \tau_1 \\ \phi_1^{-1} - 1 \end{bmatrix}, \qquad h_2(x_2) \coloneqq \begin{bmatrix} e_2^\top \\ e_3^\top \end{bmatrix} p_2, \qquad h_3(x_3) \coloneqq e_3^\top p_3,$$

$$h_i(x_i) \coloneqq 0, \quad \forall i \in \{3, \dots, N\}.$$

For this problem, we have that $\forall i$

$$
\begin{aligned}
&\nabla_{x_i} f_{ij}(x_i, x_j) \\
&= \begin{bmatrix} -\frac{1}{2}\frac{\bar{\tau}_j - \bar{\tau}_i + \frac{\|p_j - p_i\|}{c} - \phi_j^{-1} t_{ij} + \phi_i^{-1} t_i}{\sigma_{ij}^2} \frac{1}{c\|p_j - p_i\|}(p_j - p_i)^\top \\
& - \frac{\bar{\tau}_j - \bar{\tau}_i + \frac{\|p_j - p_i\|}{c} - \phi_j^{-1} t_{ij} + \phi_i^{-1} t_i}{\sigma_{ij}^2} & \frac{\bar{\tau}_j - \bar{\tau}_i + \frac{\|p_j - p_i\|}{c} - \phi_j^{-1} t_{ij} + \phi_i^{-1} t_i}{\sigma_{ij}^2} t_i \end{bmatrix} \\
&= \begin{bmatrix} -\frac{1}{2c^2\sigma_{ij}^2}(p_j - p_i)^\top - \frac{\bar{\tau}_j - \bar{\tau}_i - \phi_j^{-1} t_{ij} + \phi_i^{-1} t_i}{2c\sigma_{ij}^2\|p_j - p_i\|}(p_j - p_i)^\top \\
& -\frac{\bar{\tau}_j - \bar{\tau}_i - \phi_j^{-1} t_{ij} + \phi_i^{-1} t_i}{\sigma_{ij}^2} - \frac{\|p_j - p_i\|}{c\sigma_{ij}^2} & \frac{\bar{\tau}_j - \bar{\tau}_i - \phi_j^{-1} t_{ij} + \phi_i^{-1} t_i}{\sigma_{ij}^2} t_i + \frac{\|p_j - p_i\|}{c\sigma_{ij}^2} t_i \end{bmatrix},
\end{aligned}
$$

$$
\begin{aligned}
&\nabla_{x_j} f_{ij}(x_i, x_j) \\
&= \begin{bmatrix} \frac{1}{2c^2\sigma_{ij}^2}(p_j - p_i)^\top + \frac{\bar{\tau}_j - \bar{\tau}_i - \phi_j^{-1} t_{ij} + \phi_i^{-1} t_i}{2c\sigma_{ij}^2\|p_j - p_i\|}(p_j - p_i)^\top \\
& \frac{\bar{\tau}_j - \bar{\tau}_i - \phi_j^{-1} t_{ij} + \phi_i^{-1} t_i}{\sigma_{ij}^2} + \frac{\|p_j - p_i\|}{c\sigma_{ij}^2} & -\frac{\bar{\tau}_j - \bar{\tau}_i - \phi_j^{-1} t_{ij} + \phi_i^{-1} t_i}{\sigma_{ij}^2} t_{ij} - \frac{\|p_j - p_i\|}{c\sigma_{ij}^2} t_{ij} \end{bmatrix},
\end{aligned}
$$

$$
\begin{aligned}
&\nabla_{x_i x_i}^2 f_{ij}(x_i, x_j) \\
&= \begin{bmatrix} \frac{1}{2c^2\sigma_{ij}^2} I_3 + \frac{\bar{\tau}_j - \bar{\tau}_i - \phi_j^{-1} t_{ij} + \phi_i^{-1} t_i}{2c\sigma_{ij}^2\|p_j - p_i\|} I_3 - \frac{\bar{\tau}_j - \bar{\tau}_i - \phi_j^{-1} t_{ij} + \phi_i^{-1} t_i}{4c\sigma_{ij}^2\|p_j - p_i\|^3}(p_j - p_i)(p_j - p_i)^\top & \frac{1}{2c\sigma_{ij}^2\|p_j - p_i\|}(p_j - p_i) & -\frac{t_i}{2c\sigma_{ij}^2\|p_j - p_i\|}(p_j - p_i) \\
\frac{1}{2c\sigma_{ij}^2\|p_j - p_i\|}(p_j - p_i)^\top & \frac{1}{\sigma_{ij}^2} & -\frac{t_i}{\sigma_{ij}^2} \\
-\frac{t_i}{2c\sigma_{ij}^2\|p_j - p_i\|}(p_j - p_i)^\top & -\frac{t_i}{\sigma_{ij}^2} & \frac{t_i^2}{\sigma_{ij}^2} \end{bmatrix}
\end{aligned}
$$

$$= \begin{bmatrix} \frac{\bar{\tau}_j - \bar{\tau}_i + \frac{\|p_j - p_i\|}{c} - \phi_j^{-1} t_{ij} + \phi_i^{-1} t_i}{2c\sigma_{ij}^2 \|p_j - p_i\|} \left(I_3 - \frac{1}{2\|p_j - p_i\|^2} (p_j - p_i)(p_j - p_i)^\top \right) & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

$$+ \frac{1}{\sigma_{ij}^2} \begin{bmatrix} \frac{1}{2c\|p_j - p_i\|}(p_j - p_i) \\ 1 \\ -t_i \end{bmatrix} \begin{bmatrix} \frac{1}{2c\|p_j - p_i\|}(p_j - p_i) \\ 1 \\ -t_i \end{bmatrix}^\top ,$$

$\nabla^2_{x_j x_j} f_{ij}(x_i, x_j)$

$$= \begin{bmatrix} \frac{\bar{\tau}_j - \bar{\tau}_i + \frac{\|p_j - p_i\|}{c} - \phi_j^{-1} t_{ij} + \phi_i^{-1} t_i}{2c\sigma_{ij}^2 \|p_j - p_i\|} \left(I_3 - \frac{1}{2\|p_j - p_i\|^2} (p_j - p_i)(p_j - p_i)^\top \right) & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

$$+ \frac{1}{\sigma_{ij}^2} \begin{bmatrix} \frac{1}{2c\|p_j - p_i\|}(p_j - p_i) \\ 1 \\ -t_{ij} \end{bmatrix} \begin{bmatrix} \frac{1}{2c\|p_j - p_i\|}(p_j - p_i) \\ 1 \\ -t_{ij} \end{bmatrix}^\top ,$$

$\nabla^2_{x_i x_j} f_{ij}(x_i, x_j)$

$$= - \begin{bmatrix} \frac{\bar{\tau}_j - \bar{\tau}_i + \frac{\|p_j - p_i\|}{c} - \phi_j^{-1} t_{ij} + \phi_i^{-1} t_i}{2c\sigma_{ij}^2 \|p_j - p_i\|} \left(I_3 - \frac{1}{2\|p_j - p_i\|^2} (p_j - p_i)(p_j - p_i)^\top \right) & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

$$- \frac{1}{\sigma_{ij}^2} \begin{bmatrix} \frac{1}{2c\|p_j - p_i\|}(p_j - p_i) \\ 1 \\ -t_{ij} \end{bmatrix} \begin{bmatrix} \frac{1}{2c\|p_j - p_i\|}(p_j - p_i) \\ 1 \\ -t_i \end{bmatrix}^\top, \quad i \neq j,$$

$$\nabla^2_{x_j x_i} f_{ij}(x_i, x_j)$$

$$= - \begin{bmatrix} \frac{\bar{\tau}_j - \bar{\tau}_i + \frac{\|p_j - p_i\|}{c} - \phi_j^{-1} t_{ij} + \phi_i^{-1} t_i}{2c\sigma_{ij}^2 \|p_j - p_i\|} \Big(I_3 - \frac{1}{2\|p_j - p_i\|^2}(p_j - p_i)(p_j - p_i)^\top \Big) & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

$$- \frac{1}{\sigma_{ij}^2} \begin{bmatrix} \frac{1}{2c\|p_j - p_i\|}(p_j - p_i) \\ 1 \\ -t_i \end{bmatrix} \begin{bmatrix} \frac{1}{2c\|p_j - p_i\|}(p_j - p_i) \\ 1 \\ -t_{ij} \end{bmatrix}^\top, \quad i \neq j,$$

$$\nabla_{x_1} h_1(x_1) = I_5, \qquad \nabla_{x_2} h_2(x_2) = \begin{bmatrix} e_2^\top \\ e_3^\top \end{bmatrix}, \qquad \nabla_{x_3} h_3(x_3) = e_3^\top,$$

$$\nabla_{x_i} h_i(x_i) = 0, \quad \forall i \in \{3, \ldots, N\}, \qquad \nabla^2_{x_i x_i} h_i(x_i) = 0, \ \forall i.$$

In the absence of noise (i.e, $\bar{w}_{ij} = 0$) and for values of $p_i, p_j, \bar{\tau}_i, \bar{\tau}_j, \phi_i, \phi_j$ compatible with the measurements t_i, t_{ij}, (i.e., $\phi_j^{-1} t_{ij} - \phi_i^{-1} t_i = \bar{\tau}_j - \bar{\tau}_i + \frac{\|p_j - p_i\|}{c}$), the Hessians simplify to

$$\nabla^2_{x_i x_i} f_{ij}(x_i, x_j) = \frac{1}{\sigma_{ij}^2} \begin{bmatrix} \frac{1}{2c} s_{ij} \\ 1 \\ -t_i \end{bmatrix} \begin{bmatrix} \frac{1}{2c} s_{ij}^\top & 1 & -t_i \end{bmatrix},$$

$$\nabla^2_{x_j x_j} f_{ij}(x_i, x_j) = \frac{1}{\sigma_{ij}^2} \begin{bmatrix} \frac{1}{2c} s_{ij} \\ 1 \\ -t_{ij} \end{bmatrix} \begin{bmatrix} \frac{1}{2c} s_{ij}^\top & 1 & -t_{ij} \end{bmatrix},$$

$$\nabla^2_{x_i x_j} f_{ij}(x_i, x_j) = -\frac{1}{\sigma_{ij}^2} \begin{bmatrix} \frac{1}{2c} s_{ij} \\ 1 \\ -t_{ij} \end{bmatrix} \begin{bmatrix} \frac{1}{2c} s_{ij}^\top & 1 & -t_i \end{bmatrix},$$

$$\nabla^2_{x_j x_i} f_{ij}(x_i, x_j) = -\frac{1}{\sigma_{ij}^2} \begin{bmatrix} \frac{1}{2c} s_{ij} \\ 1 \\ -t_i \end{bmatrix} \begin{bmatrix} \frac{1}{2c} s_{ij}^\top & 1 & -t_{ij} \end{bmatrix},$$

where $s_{ij} \coloneqq \frac{1}{\|p_j - p_i\|}(p_j - p_i)$ is the unit vector pointing from p_i to p_j.

2.2 Distributed Solution to the Maximum Likelihood Estimation

2.2.1 The 1-Hop Partial Optimization

We construct a distributed algorithm for solving (2.1), where each node $i \in \mathcal{N}$ receives estimates x_j of the optimal solutions from its neighbors $j \in \mathcal{N}_i$ and, based on these estimates, computes a value for $x_i \in \mathbb{R}^{n_i}$ that minimizes only the terms in the cost function in (2.1) that depend on x_i:

$$\min_{x_i} \quad f_i(x_i) + \sum_{j \in \mathcal{N}_i} f_{ij}(x_i, x_j) + \sum_{j \in \mathcal{N} : i \in \mathcal{N}_j} f_{ji}(x_j, x_i), \tag{2.3a}$$

$$\text{subject to} \quad h_i(x_i) = 0. \tag{2.3b}$$

In practice, node $i \in \mathcal{N}$ computes values for $x_i \in \mathbb{R}^{n_i}$ and Lagrange multipliers $\lambda_j \in \mathbb{R}^{m_j}$ that satisfy the first-order necessary optimality conditions for (2.3):

$$\nabla_{x_i} f_i(x_i) + \lambda_i^\top \nabla_{x_i} h_i(x_i) + \sum_{j \in \mathcal{N}_i} \nabla_{x_i} f_{ij}(x_i, x_j) + \sum_{j \in \mathcal{N} : i \in \mathcal{N}_j} \nabla_{x_i} f_{ji}(x_j, x_i) = 0, \tag{2.4a}$$

$$h_i(x_i) = 0. \tag{2.4b}$$

If all nodes succeed in jointly satisfying (2.4), then the first-order optimality conditions for the optimal (2.1) are automatically satisfied. This observation motivates the following iterative algorithm, which is inspired by Jacobi's method for solving a system of linear equations as described in [5]. The 1-Hop Distributed Multi-Agent Maximum Likelihood Estimate (1HopDMAMLE) Algorithm utilizes the estimates from the neighboring nodes to locally solve a sequence of optimization problems. The variables $\hat{x}_i(k)$ and $\hat{\lambda}_i(k)$ should be regarded as the estimates for x_i^* and λ_i^*, respectively, at iteration k.

Algorithm 1 1HopDMAMLE Algorithm for agent i

Require: a tolerance $\delta > 0$

1: $k \leftarrow 0$

2: Initialize $\hat{x}_i(0)$ and $\hat{\lambda}_i(0)$

3: **repeat**

4: Broadcast $\hat{x}_i(k)$ to neighbors

5: $\hat{x}_i(k+1) \leftarrow \arg\min_{x_i} f_i(x_i) + \sum_{j \in \mathcal{N}_i} f_{ij}(x_i, \hat{x}_j(k)) + \sum_{j \in \mathcal{N} : i \in \mathcal{N}_j} f_{ji}(\hat{x}_j(k), x_i)$

subject to $h_i(x_i) = 0$

6: $error \leftarrow \|\hat{x}_i(k+1) - \hat{x}_i(k)\|$

7: $k \leftarrow k+1$

8: **until** $error \leqslant \delta$

return $\hat{x}_i(k)$

In this work, we restrict our attention to problems where the first-order necessary optimality conditions for the optimization problem in line 5 of Algorithm 1 and given by

$$\nabla_{x_i} f_i(\hat{x}_i(k+1)) + \hat{\lambda}_i(k+1)^\top \nabla_{x_i} h_i(\hat{x}_i(k+1)) + \sum_{j \in \mathcal{N}_i} \nabla_{x_i} f_{ij}(\hat{x}_i(k+1), \hat{x}_j(k)) + \sum_{j \in \mathcal{N} : i \in \mathcal{N}_j} \nabla_{x_i} f_{ji}(\hat{x}_j(k), \hat{x}_i(k+1)) = 0, \quad (2.5a)$$

$$h_i(\hat{x}_i(k+1)) = 0, \quad (2.5b)$$

uniquely determine $\hat{x}_i(k+1) \in \mathbb{R}^{n_i}$ and $\hat{\lambda}_i(k+1) \in \mathbb{R}^{m_i}$. Lemma 1 below shows that this will happen under mild assumptions.

2.2.2 Local Stability

To study the convergence of the Algorithm 1, we view the sequences $\hat{x}(k) \coloneqq \big(\hat{x}_1(k), \hat{x}_2(k) \ldots, \hat{x}_N(k)\big)$ and $\hat{\lambda}(k) \coloneqq \big(\hat{\lambda}_1(k), \hat{\lambda}_2(k), \ldots, \hat{\lambda}_N(k)\big)$ as the state of a discrete-time dynamical system whose dynamics are defined by (2.5) and study its local stability around an optimum $x^* \coloneqq (x_1^*, x_2^*, \ldots, x_N^*)$ for (2.1) and the corresponding Lagrange multiplier $\lambda^* \coloneqq (\lambda_1^*, \lambda_2^*, \ldots, \lambda_N^*)$.

The following result is a direct consequence of the Implicit Function Theorem [23] applied to (2.5):

Lemma 1. *Let $x^* \coloneqq (x_1^*, x_2^*, \ldots, x_N^*)$ be an optimum for (2.1) and $\lambda^* \coloneqq (\lambda_1^*, \lambda_2^*, \ldots, \lambda_N^*)$ the corresponding Lagrange multiplier. Assume that*

A1 *all the functions f_i, f_{ij}, h_i, $i \in \mathcal{N}$, $j \in \mathcal{N}_j$ are twice continuous differentiable in an open neighborhood of (x^*, λ^*); and*

A2 *the following Jacobian matrix*

$$\begin{bmatrix} F_i^* & {H_i^*}^\top \\ H_i^* & 0 \end{bmatrix} \in \mathbb{R}^{(n_i+m_i)\times(n_i+m_i)},$$

is invertible, where

$$F_i^* := \nabla^2_{x_i x_i} f_i(x_i^*) + \lambda_i^{*\top} \nabla^2_{x_i x_i} h_i(x_i^*) + \sum_{j \in \mathcal{N}_i} \nabla^2_{x_i x_i} f_{ij}(x_i^*, x_j^*) + \sum_{j \in \mathcal{N} : i \in \mathcal{N}_j} \nabla^2_{x_i x_i} f_{ji}(x_j^*, x_i^*) \in \mathbb{R}^{n_i \times n_i}$$

is a symmetric matrix and

$$H_i^* := \nabla_{x_i} h_i(x_i^*) \in \mathbb{R}^{m_i \times m_i}.$$

Then there exists an open neighborhood of (x^*, λ^*) *such that if* $\big(\hat{x}(k), \hat{\lambda}(k)\big)$ *belong to this neighborhood,* $x_i^+ := \hat{x}_i(k+1)$ *and* $\lambda^+ := \hat{\lambda}_i(k+1)$ *are uniquely defined by* (2.5) *and we have that*

$$\nabla_{x_\ell} \begin{bmatrix} x_i^+ \\ \lambda_i^+ \end{bmatrix} = \begin{bmatrix} F_i^* & H_i^{*\top} \\ H_i^* & 0 \end{bmatrix}^{-1} \begin{bmatrix} S_{i\ell}^* \\ 0 \end{bmatrix}_{(n_i+m_i) \times n_\ell}, \tag{2.6a}$$

$$\nabla_{\lambda_\ell} \begin{bmatrix} x_i^+ \\ \lambda_i^+ \end{bmatrix} = 0_{(n_i+m_i) m_\ell}, \tag{2.6b}$$

where $S^*_{i\ell} \in \mathbb{R}^{n_i \times n_\ell}$,

$$S^*_{i\ell} := \begin{cases} -\nabla^2_{x_i x_\ell} f_{i\ell}(x^*_i, x^*_\ell) - \nabla^2_{x_i x_\ell} f_{\ell i}(x^*_\ell, x^*_i), & \ell \in \mathcal{N}_i, i \in \mathcal{N}_\ell, \\ -\nabla^2_{x_i x_\ell} f_{i\ell}(x^*_i, x^*_\ell), & \ell \in \mathcal{N}_i, i \notin \mathcal{N}_\ell, \\ -\nabla^2_{x_i x_\ell} f_{\ell i}(x^*_\ell, x^*_i), & \ell \notin \mathcal{N}_i, i \in \mathcal{N}_\ell, \\ 0 & \textit{otherwise.} \end{cases}$$

□

Remark 1. For the problems discussed in Section 2.1, it can be shown that Assumption A1 of Lemma 1 holds as long as no two nodes are at the same position. Assumption A2 has simple geometric interpretations for the several localization problems. To express these conditions, we denote by $\bar{\mathcal{N}}_i$ the union of the set $\mathcal{N}_i$ of neighbors of i together with the set of nodes $j \in \mathcal{N}$ to which i is a neighbor, i.e.,

$$\bar{\mathcal{N}}_i := \mathcal{N}_i \cup \big\{ j \in \mathcal{N} : i \in \mathcal{N}_j \big\}.$$

Based on the nodes position, these conditions are as follows:

- For the relative measurements problem in Section 2.1.1 we have that the Jacobian corresponding to the reference node is

$$\begin{bmatrix} F^*_i & {H^*_i}^\top \\ H^*_i & 0 \end{bmatrix} = \begin{bmatrix} \sum_{j \in \mathcal{N}_i} \Sigma^{-1}_{ij} + \sum_{j \in \mathcal{N} : i \in \mathcal{N}_j} \Sigma^{-1}_{ji} & I_n \\ I_n & 0_{n \times n} \end{bmatrix}, \qquad i = 1,$$

and we conclude that this matrix is invertible.

For the remaining nodes in the network we have unconstrained optimizations and the Jacobian simplifies to

$$\begin{bmatrix} F_i^* \end{bmatrix} = \begin{bmatrix} \sum_{j \in \mathcal{N}_i} \Sigma_{ij}^{-1} + \sum_{j \in \mathcal{N}: i \in \mathcal{N}_j} \Sigma_{ji}^{-1} \end{bmatrix}, \qquad \forall i \in \{2, 3, \ldots, N\}.$$

Since $\Sigma_{ij} > 0 \ \forall i, j$, then we conclude that Assumption A2 holds provided that $\bar{\mathcal{N}}_i$ is not empty.

- For the range measurements problem in Section 2.1.2 with points in $\mathbb{R}^2$ and in the absence of noise, we have that for the node 1 that defines the origin of the coordinate system (i.e., $p_1 = \begin{bmatrix} 0 & 0 \end{bmatrix}^\top$), the Jacobian is

$$\begin{bmatrix} F_i^* & H_i^{*\top} \\ H_i^* & 0 \end{bmatrix} = \begin{bmatrix} \sum_{j \in \mathcal{N}_i} \frac{1}{4\sigma_{ij}^2} s_{ij} s_{ij}^\top + \sum_{j \in \mathcal{N}: i \in \mathcal{N}_j} \frac{1}{4\sigma_{ji}^2} s_{ji} s_{ji}^\top & I_2 \\ I_2 & 0_{2\times 2} \end{bmatrix}, \qquad i = 1,$$

and we note that this matrix is always invertible.

As for the node 2 that defines the x axis (i.e, p_2, such that $e_2^\top p_2 = 0$) we have the following Jacobian,

$$\begin{bmatrix} F_i^* & H_i^{*\top} \\ H_i^* & 0 \end{bmatrix} = \begin{bmatrix} \sum_{j \in \mathcal{N}_i} \frac{1}{4\sigma_{ij}^2} s_{ij} s_{ij}^\top + \sum_{j \in \mathcal{N}: i \in \mathcal{N}_j} \frac{1}{4\sigma_{ji}^2} s_{ji} s_{ji}^\top & e_2 \\ e_2^\top & 0 \end{bmatrix}, \qquad i = 2.$$

This matrix is invertible if and only if $(F_i)_{11} \neq 0$. Thus Assumption A2 is satisfied by imposing that node 1 is in the neighborhood of node 2, i.e., $\{1\} \in \bar{\mathcal{N}}_2$.

Finally, for the remaining nodes in the network the optimizations are unconstrained and the Jacobian reduces to

$$\begin{bmatrix} F_i^* \end{bmatrix} = \begin{bmatrix} \sum_{j\in\mathcal{N}_i} \frac{1}{4\sigma_{ij}^2} s_{ij}s_{ij}^\top + \sum_{j\in\mathcal{N}:i\in\mathcal{N}_j} \frac{1}{4\sigma_{ji}^2} s_{ji}s_{ji}^\top \end{bmatrix}, \qquad \forall i \in \{3,4,\ldots,N\}.$$

For this matrix, we conclude from Lemma 6 that Assumption A2 holds provided that $\bar{\mathcal{N}}_i$ contains at least two nodes such that these points are not co-linear with p_i.

- For the range measurements problem in Section 2.1.2 with points in $\mathbb{R}^3$ and in the absence of noise, we have the following Jacobian for the node 1 that defines the origin of the coordinate system (i.e, $p_1 = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}^\top$).

$$\begin{bmatrix} F_i^* & H_i^{*\top} \\ H_i^* & 0 \end{bmatrix} = \begin{bmatrix} \sum_{j\in\mathcal{N}_i} \frac{1}{4\sigma_{ij}^2} s_{ij}s_{ij}^\top + \sum_{j\in\mathcal{N}:i\in\mathcal{N}_j} \frac{1}{4\sigma_{ji}^2} s_{ji}s_{ji}^\top & I_3 \\ I_3 & 0_{3\times 3} \end{bmatrix}, \qquad i = 1,$$

and we note that this matrix is always invertible.

As for the node 2 that defines the x axis (i.e, $e_2^\top p_2 = e_3^\top p_2 = 0$) we have that,

$$\begin{bmatrix} F_i^* & H_i^{*\top} \\ H_i^* & 0 \end{bmatrix} = \begin{bmatrix} \sum_{j\in\mathcal{N}_i} \frac{1}{4\sigma_{ij}^2} s_{ij}s_{ij}^\top + \sum_{j\in\mathcal{N}:i\in\mathcal{N}_j} \frac{1}{4\sigma_{ji}^2} s_{ji}s_{ji}^\top & \begin{bmatrix} e_2 & e_3 \end{bmatrix} \\ \begin{bmatrix} e_2^\top \\ e_3^\top \end{bmatrix} & 0_{2\times 2} \end{bmatrix}, \qquad i = 2.$$

Note that this matrix is invertible provided that $(F_i)_{11} \neq 0$. A sufficient condition for that is to have node 1 connected to node 2, $\{1\} \in \bar{\mathcal{N}}_2$.

For the node 3 that defines the second axis (i.e, $e_3^\top p_3 = 0$) we have that,

$$\begin{bmatrix} F_i^* & H_i^{*\top} \\ H_i^* & 0 \end{bmatrix} = \begin{bmatrix} \sum_{j\in\mathcal{N}_i} \frac{1}{4\sigma_{ij}^2} s_{ij}s_{ij}^\top + \sum_{j\in\mathcal{N}:i\in\mathcal{N}_j} \frac{1}{4\sigma_{ji}^2} s_{ji}s_{ji}^\top & e_3 \\ e_3^\top & 0 \end{bmatrix}, \qquad i = 3.$$

This matrix is invertible if and only if $(F_i^*)_{11}(F_i^*)_{22} - (F_i^*)_{12}(F_i^*)_{21} \neq 0$. Denote $w_{ij} \in \mathbb{R}^2$ the first and second component of the unit vector s_{ij} such that $w_{ij} = \begin{bmatrix} (s_{ij})_1 & (s_{ij})_2 \end{bmatrix}^\top$. Then we have that

$$\sum_{j\in\mathcal{N}_i} \frac{1}{4\sigma_{ij}^2} w_{ij}w_{ij}^\top + \sum_{j\in\mathcal{N}:i\in\mathcal{N}_j} \frac{1}{4\sigma_{ji}^2} w_{ji}w_{ji}^\top = \begin{bmatrix} (F_i^*)_{11} & (F_i^*)_{12} \\ (F_i^*)_{21} & (F_i^*)_{22} \end{bmatrix}.$$

Thus, from Lemma 6, we conclude that Assumtion A2 holds if $\bar{\mathcal{N}}_i$ contains at least two nodes with points that are not co-linear with p_3 in the $x - y$ plane.

Finally, for the remaining nodes, a similar argument from the $\mathbb{R}^2$ case follows and we conclude from Lemma 6 that Assumption A2 holds provided that $\bar{\mathcal{N}}_i$ contains at least three points that are not co-linear with p_i $\forall i > 3$.

- For the pseudo-range measurements problem in Section 2.1.3 with points in $\mathbb{R}^2$ and in the absence of noise, we have that for the node 1 that defines the origin of the coordinate system and the time offset reference ($p_1 = \begin{bmatrix} 0 & 0 \end{bmatrix}^\top$, $\tau_1 = 0$), the Jacobian is of the form,

$$\begin{bmatrix} F_i^* & H_i^{*\top} \\ H_i^* & 0 \end{bmatrix} = \begin{bmatrix} \sum_{j \in \mathcal{N}_i} \frac{1}{\sigma_{ij}^2} v_{ij} v_{ij}^\top + \sum_{j \in \mathcal{N} : i \in \mathcal{N}_j} \frac{1}{\sigma_{ji}^2} v_{ji} v_{ji}^\top & I_3 \\ I_3 & 0_{3\times 3} \end{bmatrix}, \qquad i = 1,$$

and we conclude that the matrix is invertible.

For the node 2 that defines the x axis ($e_2^\top p_2 = 0$) we have that

$$\begin{bmatrix} F_i^* & H_i^{*\top} \\ H_i^* & 0 \end{bmatrix} = \begin{bmatrix} \sum_{j \in \mathcal{N}_i} \frac{1}{\sigma_{ij}^2} v_{ij} v_{ij}^\top + \sum_{j \in \mathcal{N} : i \in \mathcal{N}_j} \frac{1}{\sigma_{ji}^2} v_{ji} v_{ji}^\top & e_2^\top \\ e_2 & 0 \end{bmatrix}, \qquad i = 2.$$

This matrix is invertible if and only if $(F_i)_{11}(F_i)_{33} - (F_i)_{13}(F_i)_{31} \neq 0$. Again, denote $w_{ij} \in \mathbb{R}^2$ the vector formed from the first and the third component of the vector v_{ij}, $w_{ij} = \begin{bmatrix} (v_{ij})_1 & (v_{ij})_3 \end{bmatrix}^\top = \begin{bmatrix} (v_{ij})_1 & 1 \end{bmatrix}^\top$. Then we have that the Jacobian is invertible if and only if the submatrix

$$\sum_{j \in \mathcal{N}_i} \frac{1}{\sigma_{ij}^2} w_{ij} w_{ij}^\top + \sum_{j \in \mathcal{N} : i \in \mathcal{N}_j} \frac{1}{\sigma_{ji}^2} w_{ji} w_{ji}^\top = \begin{bmatrix} (F_i^*)_{11} & (F_i^*)_{13} \\ (F_i^*)_{31} & (F_i^*)_{33} \end{bmatrix}, \qquad i = 2,$$

is invertible. From Lemma 6 we conclude that Assumption A2 holds if $\bar{\mathcal{N}}_2$ contains at least two nodes such that when projected in the x axis they are different.

For the remaining nodes, Assumption A2 holds provided that there exists 2 nodes in $\bar{\mathcal{N}}_i,\ \forall i > 2$ that are linearly independent in the $x - y$ plane and a third node that does not have the same coordinates as these 2 nodes.

- For the pseudo-range measurements problem in Section 2.1.3 with points in $\mathbb{R}^3$ and in the absence of noise, we have that for the node 1 that defines the origin and the time offset reference ($p_1 = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}^\top,\ \tau_1 = 0$), the Jacobian is

$$\begin{bmatrix} F_i^* & H_i^{*\top} \\ H_i^* & 0 \end{bmatrix} = \begin{bmatrix} \sum_{j \in \mathcal{N}_i} \frac{1}{\sigma_{ij}^2} v_{ij} v_{ij}^\top + \sum_{j \in \mathcal{N} : i \in \mathcal{N}_j} \frac{1}{\sigma_{ji}^2} v_{ji} v_{ji}^\top & I_4 \\ I_4 & 0_{4\times 4} \end{bmatrix}, \qquad i = 1,$$

which is invertible.

As for the node 2 that defines the x axis ($e_2^\top p_2 = e_3^\top p_2 = 0$), the Jacobian is

$$\begin{bmatrix} F_i^* & H_i^{*\top} \\ H_i^* & 0 \end{bmatrix} = \begin{bmatrix} \sum_{j \in \mathcal{N}_i} \frac{1}{\sigma_{ij}^2} v_{ij} v_{ij}^\top + \sum_{j \in \mathcal{N} : i \in \mathcal{N}_j} \frac{1}{\sigma_{ji}^2} v_{ji} v_{ji}^\top & \begin{bmatrix} e_2 & e_3 \end{bmatrix} \\ \begin{bmatrix} e_2^\top \\ e_3^\top \end{bmatrix} & 0_{2\times 2} \end{bmatrix}, \qquad i = 2.$$

This matrix is invertible if and only if $(F_i^*)_{11}(F_i^*)_{44} - (F_i^*)_{14}(F_i^*)_{41} \neq 0$. Let $w_{ij} \in \mathbb{R}^2$ be the vector formed from the first and fourth component of the vector s_{ij}, $w_{ij} := \begin{bmatrix} (v_{ij})_1 & (v_{ij})_2 \end{bmatrix}^\top = \begin{bmatrix} (v_{ij})_1 & 1 \end{bmatrix}^\top$. Then, similarly from the analysis for the previous scenario, we have that

$$\sum_{j\in\mathcal{N}_i}\frac{1}{\sigma_{ij}^2}w_{ij}w_{ij}^\top+\sum_{j\in\mathcal{N}:i\in\mathcal{N}_j}\frac{1}{\sigma_{ji}^2}w_{ji}w_{ji}^\top=\begin{bmatrix}(F_i^*)_{11} & (F_i^*)_{14}\\ (F_i^*)_{41} & (F_i^*)_{44}\end{bmatrix},\qquad i=2,$$

and we conclude that Assumption A2 holds if $\bar{\mathcal{N}}_2$ contains at least two nodes that when projected in the x axis they are not equal.

For the node 3 that defines the second axis ($e_3^\top p_3=0$) we have that the associate Jacobian is

$$\begin{bmatrix}F_i^* & H_i^{*\top}\\ H_i^* & 0\end{bmatrix}=\begin{bmatrix}\sum_{j\in\mathcal{N}_i}\frac{1}{\sigma_{ij}^2}v_{ij}v_{ij}^\top+\sum_{j\in\mathcal{N}:i\in\mathcal{N}_j}\frac{1}{\sigma_{ji}^2}v_{ji}v_{ji}^\top & e_3^\top\\ e_3 & 0\end{bmatrix},\qquad i=3.$$

Defining $w_{ij}\coloneqq\begin{bmatrix}(v_i)_1 & (v_i)_2 & (v_i)_3\end{bmatrix}^\top=\begin{bmatrix}(v_i)_1 & (v_i)_2 & 1\end{bmatrix}^\top$ we have that the Jacobian is invertible if and only if the following submatrix of F_i,

$$\sum_{j\in\mathcal{N}_i}\frac{1}{\sigma_{ij}^2}w_{ij}w_{ij}^\top+\sum_{j\in\mathcal{N}:i\in\mathcal{N}_j}\frac{1}{\sigma_{ji}^2}w_{ji}w_{ji}^\top=\begin{bmatrix}(F_i^*)_{11} & (F_i^*)_{12} & (F_i^*)_{14}\\ (F_i^*)_{21} & (F_i^*)_{22} & (F_i^*)_{24}\\ (F_i^*)_{41} & (F_i^*)_{42} & (F_i^*)_{44}\end{bmatrix},\qquad i=3,$$

is invertible. Thus, from Lemma 6 we conclude that Assumption A2 holds if there exists at least 2 nodes in $\bar{\mathcal{N}}_3$ that are linearly independent when projected to the

$x-y$ plane, and a third node such that when projected in the same plane does not coincide with the previous two nodes' projections.

Finally, for the remaining nodes in the network, we conclude from Lemma 6 that Assumption A2 holds provided that $\bar{\mathcal{N}}_i,\ \forall i > 3$ contains at least 3 points that are linearly independent and a fourth point that is not equal to these previous points.

- For the pseudo-range measurements with clock drift and bias problem in Section 2.1.4 with points in $\mathbb{R}^2$ and in the absence of noise, we have that for the node 1 that defines the origin of the coordinate system and the clock reference ($p_1 = \begin{bmatrix} 0 & 0 \end{bmatrix}^\top$,$\tau_1 = 0$, $\phi_1^{-1} = 1$) the Jacobian is

$$\begin{bmatrix} F_i^* & H_i^{*\top} \\ H_i^* & 0 \end{bmatrix} = \begin{bmatrix} \sum_{j\in\mathcal{N}_i} \frac{1}{\sigma_{ij}^2} v_{ij} v_{ij}^\top + \sum_{j\in\mathcal{N}: i\in\mathcal{N}_j} \frac{1}{\sigma_{ji}^2} v_{ji} v_{ji}^\top & I_4 \\ I_4 & 0_{4\times 4} \end{bmatrix}, \qquad i = 1,$$

which is invertible. For the node 2 that defines the x axis ($e_2^\top p_2 = 0$) the Jacobian is of the form

$$\begin{bmatrix} F_i^* & H_i^{*\top} \\ H_i^* & 0 \end{bmatrix} = \begin{bmatrix} \sum_{j\in\mathcal{N}_i} \frac{1}{\sigma_{ij}^2} v_{ij} v_{ij}^\top + \sum_{j\in\mathcal{N}: i\in\mathcal{N}_j} \frac{1}{\sigma_{ji}^2} v_{ji} v_{ji}^\top & e_2 \\ e_2^\top & 0 \end{bmatrix}, \qquad i = 2.$$

Defining $w_{ij} \coloneqq \begin{bmatrix} (v_i)_1 & (v_i)_3 & (v_i)_3 \end{bmatrix}^\top = \begin{bmatrix} (v_i)_1 & 1 & t_i \end{bmatrix}^\top$ we have that the Jacobian is invertible if and only if the following submatrix

$$\sum_{j\in\mathcal{N}_i}\frac{1}{\sigma_{ij}^2}w_{ij}w_{ij}^\top+\sum_{j\in\mathcal{N}:i\in\mathcal{N}_j}\frac{1}{\sigma_{ji}^2}w_{ji}w_{ji}^\top=\begin{bmatrix}(F_i^*)_{11} & (F_i^*)_{13} & (F_i^*)_{14}\\ (F_i^*)_{31} & (F_i^*)_{33} & (F_i^*)_{34}\\ (F_i^*)_{41} & (F_i^*)_{43} & (F_i^*)_{44}\end{bmatrix}, \qquad i=2,$$

is invertible. There are many ways Assumption A2 holds in light of Lemma 6. One such way is to have node 2 connected to at least two other nodes, such that one is an outgoing neighbor (say node i) and the other node is both an incoming and an outgoing neighbor (say node j). For these nodes it must be true that their clock measurements of node 2 is different from the measurement received from the clock at node j (i.e, $t_2 \neq t_{2j}$), and that the two neighbor nodes have different projection in the x axis. □

Lemma 1 enables us to compute the local linearization of the discrete-time dynamical system whose dynamics are defined by (2.5) around an optimum $x^* := (x_1^*, x_2^*, \ldots, x_N^*)$ for (2.1) with Lagrange multipliers $\lambda^* := (\lambda_1^*, \lambda_2^*, \ldots, \lambda_N^*)$. Denoting by $\delta x := (\delta x_1, \delta x_2, \ldots, \delta x_N)$ and $\delta\lambda := (\delta\lambda_1, \delta\lambda_2, \ldots, \delta\lambda_N)$ the perturbations of the state with respect to the equilibrium point (x^*, λ^*), under the assumptions of Lemma 1 we conclude from (2.6a) that the next-state vector $(\delta x^+, \delta\lambda^+)$ is uniquely defined by the following system of equations on the unknowns δx_i^+ and $\delta\lambda_i^+$.

$$\begin{bmatrix} F_i^* & H_i^{*\top} \\ H_i^* & 0 \end{bmatrix} \begin{bmatrix} \delta x_i^+ \\ \delta\lambda_i^+ \end{bmatrix} = \begin{bmatrix} \sum_{\ell\in\mathcal{N}_i} S_{i\ell}^* \delta x_\ell \\ 0 \end{bmatrix}, \quad \forall i \in \mathcal{N}.$$

In vector form, we can express the dynamical system for all the agents as

$$\begin{bmatrix} F^* & H^{*\top} \\ H^* & 0 \end{bmatrix} \begin{bmatrix} \delta x^+ \\ \delta\lambda^+ \end{bmatrix} = \begin{bmatrix} S^* \delta x \\ 0 \end{bmatrix},$$

where

$$F^* \coloneqq \operatorname{diag}(F_1^*, F_2^*, \ldots, F_N^*) \in \mathbb{R}^{n\times n}, \qquad n \coloneqq \sum_{i\in\mathcal{N}} n_i,$$

$$H^* = \operatorname{diag}(H_1^*, H_2^*, \ldots, H_N^*) \in \mathbb{R}^{m\times m}, \quad m \coloneqq \sum_{i\in\mathcal{N}} m_i,$$

$$S^* \coloneqq \left[S_{i\ell}^*\right]_{i\in\mathcal{N},\ell\in\mathcal{N}}, \in \mathbb{R}^{n\times n}.$$

The next result provides a sufficient condition for the local stability of this system.

Theorem 1. *Suppose that the assumptions of Lemma 1 hold and that there exists a scalar $\sigma \in \mathbb{R}$ such that,*

$$F^* + \sigma H^{*\top} H^* - \frac{1}{2}(S^{*\top} + S^*) > 0, \tag{2.7a}$$

$$F^* + \sigma H^{*\top} H^* + \frac{1}{2}(S^{*\top} + S^*) > 0. \tag{2.7b}$$

Then the optimum (x^*, λ^*) *for* (2.1) *is a locally asymptotically stable equilibrium point of the discrete-time dynamical system whose dynamics are defined by* (2.5). □

This result is a direct consequence of Lemma 1 and the following lemma.

Lemma 2. *Consider a discrete-time linear time-invariant system*

$$z^+ = Az, \tag{2.8}$$

where, for every $z \in \mathbb{C}^n$*, the next state* $z^+ \in \mathbb{C}^n$ *can be uniquely determined by the following equation*

$$\exists u \in \mathbb{C}^m : \begin{bmatrix} F & H^\top \\ H & 0_{m\times m} \end{bmatrix} \begin{bmatrix} z^+ \\ u \end{bmatrix} = \begin{bmatrix} Sz \\ 0_{m\times 1} \end{bmatrix}$$

for appropriate matrices $F = F^\top, S \in \mathbb{R}^{n\times n}$*,* $H \in \mathbb{R}^{m\times n}$ *such that there exists a scalar* $\sigma \in \mathbb{R}$

$$F + \sigma H^\top H - \frac{1}{2}(S^\top + S) > 0, \tag{2.9a}$$

$$F + \sigma H^\top H + \frac{1}{2}(S^\top + S) > 0. \tag{2.9b}$$

Then A *is Schur and the original system* (2.8) *is asymptotically stable.* □

Proof of Lemma 2. Consider an eigenvalue $\lambda \in \mathbb{C}\backslash\{0\}$ and the corresponding eigenvector

$v \in \mathbb{C}^n$ of A. Under the lemma's hypothesis,

$$Av = \lambda v \quad \Leftrightarrow \quad \exists u \in \mathbb{C}^m : \begin{bmatrix} F & H^\top \\ H & 0_{m\times m} \end{bmatrix} \begin{bmatrix} \lambda v \\ u \end{bmatrix} = \begin{bmatrix} Sv \\ 0_{m\times 1} \end{bmatrix}$$

$$\Leftrightarrow \quad \exists u \in \mathbb{C}^m : \lambda F v + D^\top u = Sv,\ \lambda H v = 0.$$

$$\Leftrightarrow \quad \exists u \in \mathbb{C}^m : \lambda F v + \lambda\sigma H^\top H v + H^\top u = Sv,\ \lambda H v = 0.$$

We thus conclude that

$$\lambda v^\dagger (F + \sigma H^\top H) v + v^\dagger H^\top u = v^\dagger S v,\ u^\dagger H v = 0 \ \Rightarrow\ \lambda v^\dagger (F + \sigma H^\top H) v = v^\dagger S v, \quad (2.10)$$

where $v^\dagger$ and $u^\dagger$ denote the complex conjugate transpose of v and u, respectively. On the other hand, from (2.9) we have that

$$v^\dagger (F + \sigma H^\top H) v > \frac{1}{2} v^\dagger (S^\top + S) v = v^\dagger S v, \quad (2.11a)$$

$$v^\dagger (F + \sigma H^\top H) v > -\frac{1}{2} v^\dagger (S^\top + S) v = -v^\dagger S v, \quad (2.11b)$$

which implies that $v^\dagger (F + \sigma H^\top H) v > 0$. This allow us to conclude from (2.10) and (2.11) that

$$\begin{aligned} \lambda v^\dagger (F + \sigma H^\top H) v = v^\dagger S v \quad &\Rightarrow \quad \lambda = \frac{v^\dagger S v}{v^\dagger (H + \sigma H^\top H) v}, \\ v^\dagger (F + \sigma H^\top H) v > v^\dagger S v \quad &\Rightarrow \quad \frac{v^\dagger S v}{v^\dagger (F + \sigma H^\top H) v} < 1, \end{aligned}$$

$$v^\dagger(F+\sigma H^\top H)v > -v^\dagger S v \qquad \Rightarrow \quad \frac{v^\dagger S v}{v^\dagger(F+\sigma H^\top H)v} > -1.$$

Therefore λ must be a real number in the (open) interval $(-1, 1)$. ■

Remark 2. The sufficient condition for local asymptotic stability presented in Theorem 1 and given by the matrix inequalities in (2.7) can be verified for the problems related to localization discussed in this chapter.

Consider the localization problem based on relative position measurements described in Section 2.1.1. We assume that the independent zero-mean Gaussian noises w_{ij} that corrupts the pairwise node measurements have the same co-variance matrix $\Sigma_{ij} = \Sigma > 0, \ \forall i, j$. Then,

$$F_i^* = 2d_i\Sigma^{-1}, \qquad \forall i, \qquad H_1^* = I_n, \qquad H_i^* = 0_n, \quad \forall i \in \{2, 3, \dots, N\},$$

where $d_i := |\mathcal{N}_i|$ is the degree of node i and we represent node 1 as the reference node. We consider that the graph spanned by the communication between the N nodes is connected and undirected so the $S_{i\ell}^*$ matrices reduce to

$$S_{i\ell}^* = \begin{cases} 2\Sigma^{-1} & \ell \in \mathcal{N}_i, \\ 0 & \text{otherwise.} \end{cases}$$

Then we have that for all the nodes

$$F^* = 2D \otimes \Sigma^{-1}, \qquad H^{*\top} = \begin{bmatrix} I_n & 0 \cdots & 0 \end{bmatrix}, \qquad S^* = 2A_d \otimes \Sigma^{1},$$

where $D \coloneqq \operatorname{diag}(d_1, d_2, \ldots, d_N) \in \mathbb{R}^{N\times N}$ is the degree matrix and $A_d \in \mathbb{R}^{N\times N}$ the adjacency matrix with $(A_d)_{ij} = 1$ if there exists an edge between node i and node j and zero otherwise.

For this problem, the inequalities in (2.7) are

$$\begin{aligned} F^* + \sigma H^{*\top} H^* \pm \frac{1}{2}(S^{*\top} + S^*) &= 2D \otimes \Sigma^{-1} + \sigma \begin{bmatrix} I_n & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \pm 2A_d \otimes \Sigma^{-1} \\ &= 2(D \pm A_d) \otimes \Sigma^{-1} + \sigma \begin{bmatrix} I_n & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}. \end{aligned}$$

Since the graph is connected one can show from the Gersgorin disc theorem [25] that $D \pm A_d \geqslant 0$ where for a given vector $v^\top \coloneqq \begin{bmatrix} v_1, v_2, \ldots, v_N \end{bmatrix}$ in the kernel of $D \pm A_d$, we have that $|v_1| = |v_2| = \cdots = |v_N|$. Hence for any $\sigma > 0$ the following inequality holds

$$2(D \pm A_d) \otimes \Sigma^{-1} + \sigma \begin{bmatrix} I_n & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} > 0.$$

We conclude from Theorem 1 that the estimates converge to an optimal solution for (2.1).

□

2.3 Numerical Example

To illustrate the proposed algorithm and the theoretical results obtained in the previous section, we present an example of node localization for range-based measurements in the $x - y$ plane, as described in Section 2.1.2.

Inspired by the Henneberg construction of rigid frameworks [44], we generate random networks by successively adding a node at a random position to an existing rigid framework. The new node is connected using bidirectional edges with two existing nodes such that these three nodes are not co-linear. The resulting graph spanned by these connections is connected and undirected.

We have generated a large number of rigid frameworks using the procedure described above and verified that the corresponding matrices F^*, H^*, and S^* verified the conditions in Theorem 1 for $\sigma = 1$. We thus conjecture that these conditions hold generically (perhaps excluding singular configurations) for rigid frameworks. The investigation of this conjecture is a potential future research direction.

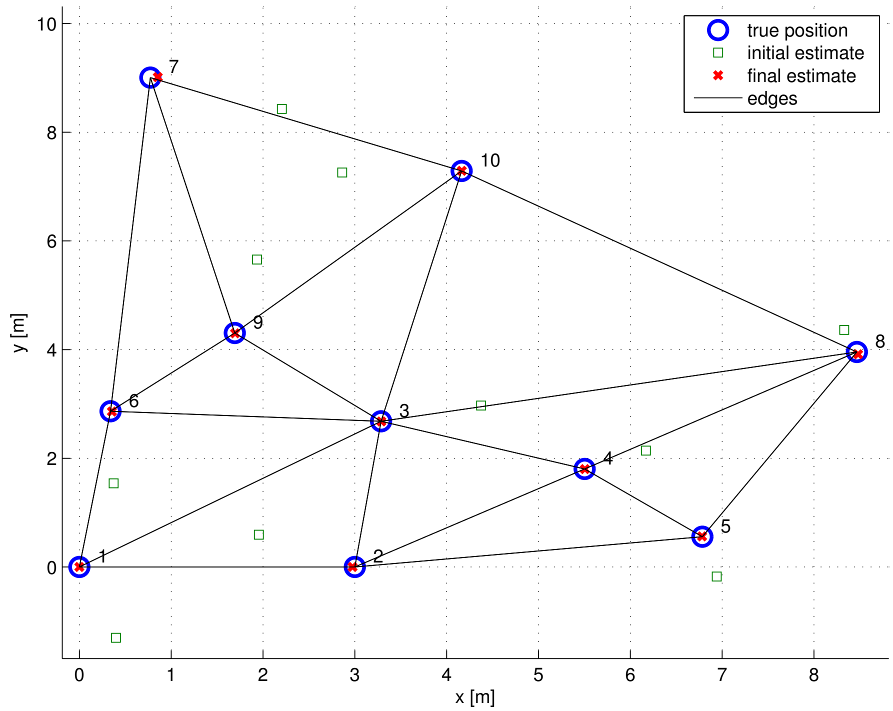


Figure 2.1: Sensor network with 10 randomly distribute nodes. The nodes share information about their current estimates and the noisy range measurements with a limited number of neighbors, represented by the edge connection. The initial estimates are random.

One such network consisting of 10 randomly distributed nodes in the $x - y$ plane is shown in Figure 2.1. Figure 2.2 shows a typical evolution of the local cost function and the estimation error for two nodes, as a function of the iteration number of the Jacobi algorithm described in Section 2.2.1 starting with a random initialization for the node estimate within a ball centered at their true positions. The interior-point method was used to solve each optimization step. We observe that node 6, which is one hop away from the reference node 1, converges faster than node 10, which is 2 hops away from the reference node 2. This is to be expected, because the convergence of the reference nodes is faster than the rest of the network, improving the speed of convergence of its direct neighbors.

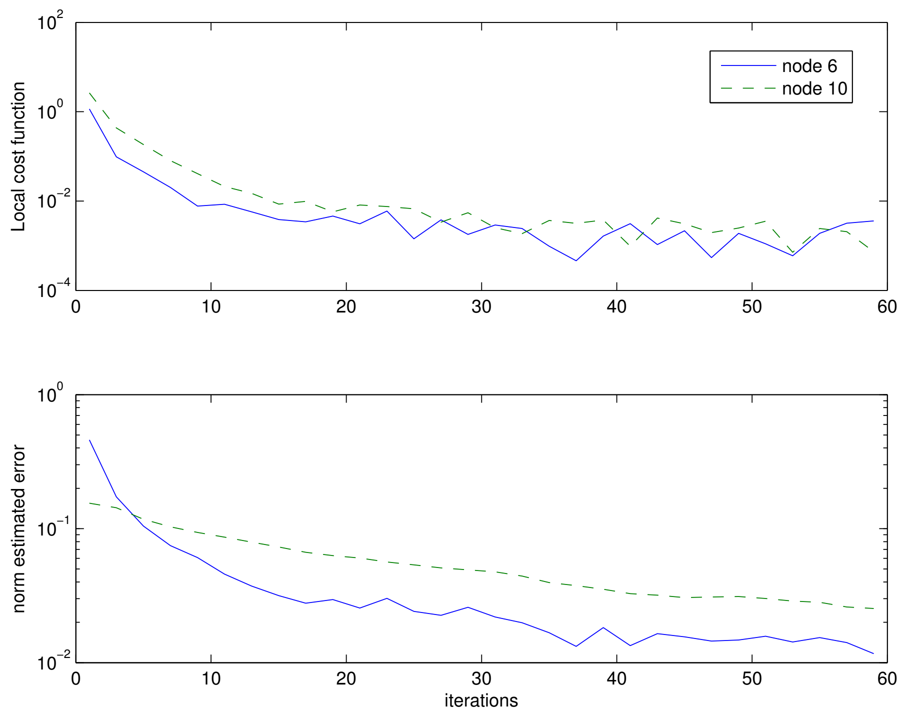


Figure 2.2: Cost function and error evolution for two nodes. The dashed line represents a node that is more distant to the reference nodes than the node represented by a solid line.

2.4 Experimental Evaluation

We evaluated experimentally the performance of the proposed algorithm for a localization and clock synchronization problem in the $x-y-z$ space with pseudo-range measurements with clock drifts and biases as described in Section 2.1.4 [2]. The setup is shown in Figure 2.3 and we used a custom ultra-wideband (UWB) Radio Frequency (RF) test bed

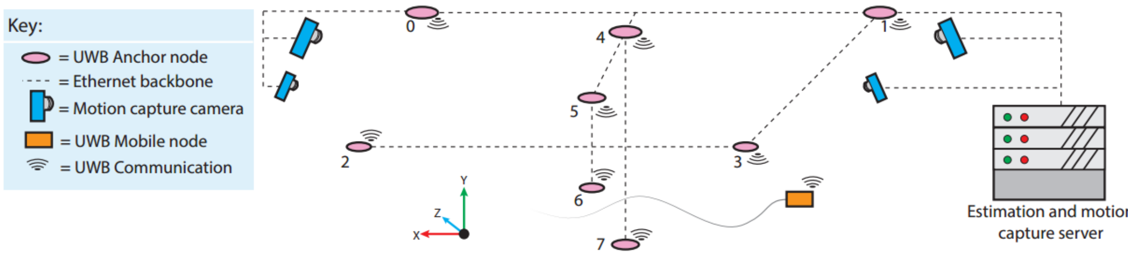


Figure 2.3: Experimental setup overview, including, UWB Anchor nodes, motion capture cameras, and mobile quadrotor UWB nodes

based on the DecaWave DW1000 IR-UWB radio [14].

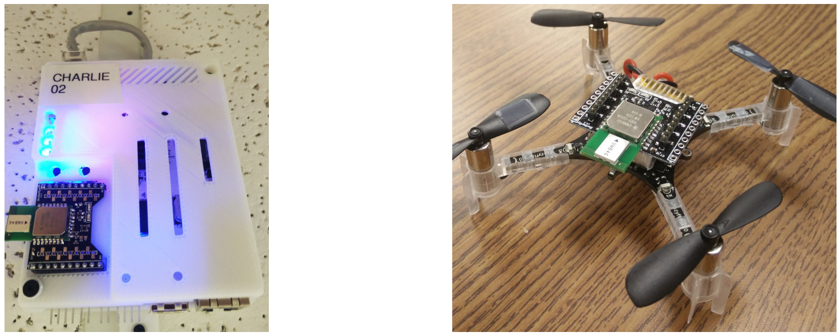
The main components of the test bed are:

- **Static nodes**: we installed eight UWB radio devices that act as static nodes in different positions in a $10{\times}9{\times}3\ m^3$ room. Six nodes were placed on the ceiling (roughly 2.5 m high) and two were placed at waist height (about 1 m) to better disambiguate positions in the vertical z axis. Each anchor node is connected to an Ethernet backbone both for power and for communication to a central server. Figure 2.4a shows one of such devices.

- **Mobile nodes**: these are battery powered devices based on the CrazyFlie 2.0 quadcopter [6] equipped with the same DW1000 radio as shown in Figure 2.4b.

- **Motion capture system**: comprised of a set of cameras capable of 3D rigid body position measurement with less than 0.5 mm accuracy used for ground truth comparison.

- **Centralized server**: this unit is utilized for aggregation of the nodes timing information and ground truth position estimates given from the motion capture



(a) Static node (b) Modile node

Figure 2.4: Types of nodes utilized in the experimental test bed.(a) Ceiling-mounted node with DW1000 UWB radio in 3D-printed enclosure. (b) CrazyFlie 2.0 quadrotor helicopter with the same UWB radio.

cameras.

The measurements collected are time stamped messages of transmission and reception time exchanged between appropriate pair of sensors. The noise is assumed to be Gaussian and the propagation velocity of radio is taken to be the speed of light in vacuum. The distributed localization experiments were performed considering mobile and static nodes.

2.4.1 Static Node Localization and Clock Synchronization

In our first experiment we considered a communication network represented by a complete graph where all the nodes exchange message among themselves. We compare our algorithm (DOPT) with three other established methods found in the literature: a distributed Kalman filter (DKAL) as detailed in [9], a distributed Kalman filter for large-scale systems (DKALarge) described in [28], and a standard centralized Kalman filter (CKAL).

Figure 2.5 illustrates the evolution of the mean localization error for the duration of the experiment. We observe that the estimates decay faster when utilizing our algorithm if compared to the other distributed methods. The final localization error for the individual nodes is shown in Table 2.1. Comparing the mean and the standard deviation for each of the methods we conclude that the method proposed in this chapter outperforms the other distributed approaches tested.

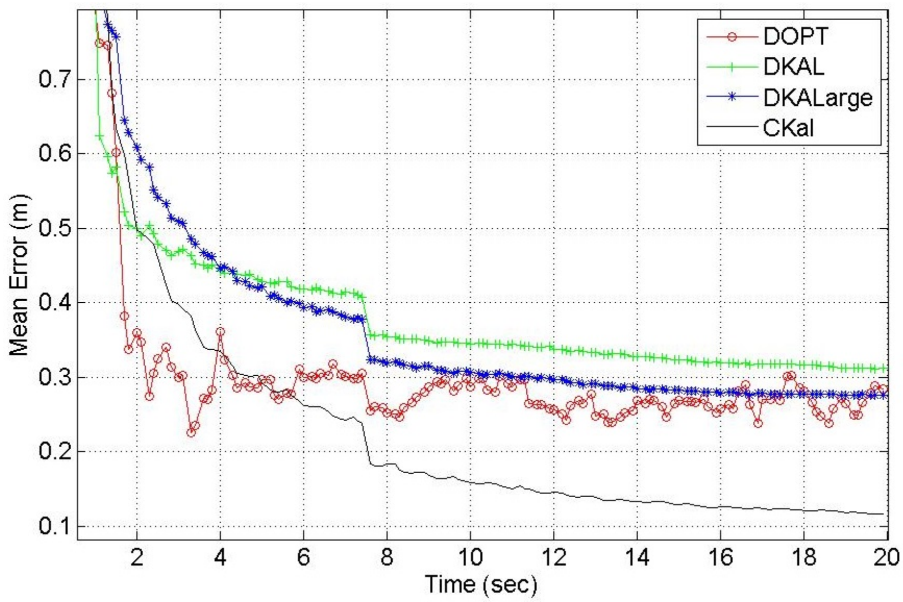


Figure 2.5: Average localization error for a fully connected network experiment with static node comparing the four different approaches.

Table 2.1: Localization error (in meters) for a fully connected network experiment with static nodes. Due to space limitation we omit the results for node 5 and node 6.

Algorithm	node 0	node 1	node 2	node 3	node 4	node 7	mean	std
DKAL	0.518	0.189	0.638	0.228	0.021	0.336	0.311	0.209
DKALarge	0.232	0.536	0.394	0.318	0.093	0.418	0.330	0.137
DOPT	0.402	0.202	0.530	0.193	0.156	0.397	0.299	0.133
CKAL	0.205	0.189	0.208	0.147	0.218	0.143	0.169	0.047

We measure the accuracy of the estimate of the clock parameters (drifts and bias) in terms of the synchronization error. Node 0 is chosen as the reference node ($\phi_0 =$

1 and $\tau_0 = 0$) so the remaining nodes estimate their clock parameters with respect to this reference. Table 2.2 summarizes the synchronization error obtained at the end of the experiment. The distributed optimization method proposed in this chapter has a significant lower mean compared to the other distributed approaches, but results in estimation error with higher standard deviation than the DKLarge method.

Table 2.2: Synchronization error (in μ seconds) of different nodes with respect to node 0. Due to space limitation we omit the results for node 5 and node 6.

Algorithm	node 1	node 2	node 3	node 4	node 7	mean	std
DKAL	0.807	9.088	1.868	2.332	5.342	5.000	3.502
DKALarge	5.223	6.448	5.203	5.339	4.222	5.087	1.036
DOPT	2.15	0.891	2.090	2.343	3.293	2.520	1.391
CKAL	1.362	2.045	1.440	1.517	0.708	1.304	0.617

2.4.2 Mobile Node Localization

In our second experiment we considered a heterogeneous network containing both static and mobile nodes. To the eight static nodes described in the previous experiment, we add a mobile node represented by a quadrotor flying at different directions and speeds.

Figure 2.6, 2.7, and 2.8 show the results of a 2 minutes flight mission experiment of localization using the DKAL, DKALarge, and DOPT, repectively. The DKAL achieved the best localization estimation with RMSE of 75 *cm*, followed by the DKLarge with 89 *cm*, and DOPT with 116 *cm*. The approaches based on Kalman filter outperformed the method proposed in this chapter since the DOPT does not take into account any dynamics in its formulation. We also note from the plots that when the mobile agent is in a location that is more central to the network (i.e., closer to the centroid defined by the mean of the nodes' positions) they are more likely to have a better estimate of their current position.

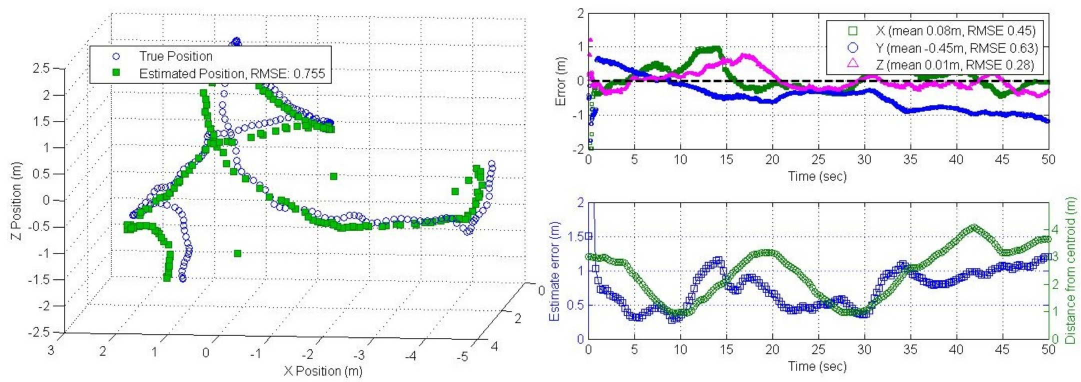


Figure 2.6: Localization errors for DKAL in 3D for a single mobile node. Spatial errors (left) are shown with corresponding per-axis errors by time (top right). Additionally, the error is plotted against the mobile nodes distance from the network centroid (bottom right).

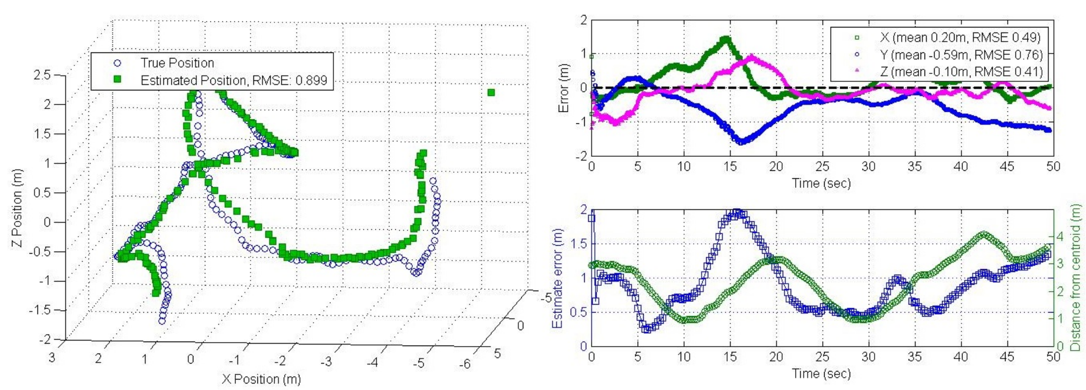


Figure 2.7: Localization errors for DKALarge in 3D for a single mobile node. Spatial errors (left) are shown with corresponding per-axis errors by time (top right). Additionally, the error is plotted against the mobile nodes distance from the network centroid (bottom right).

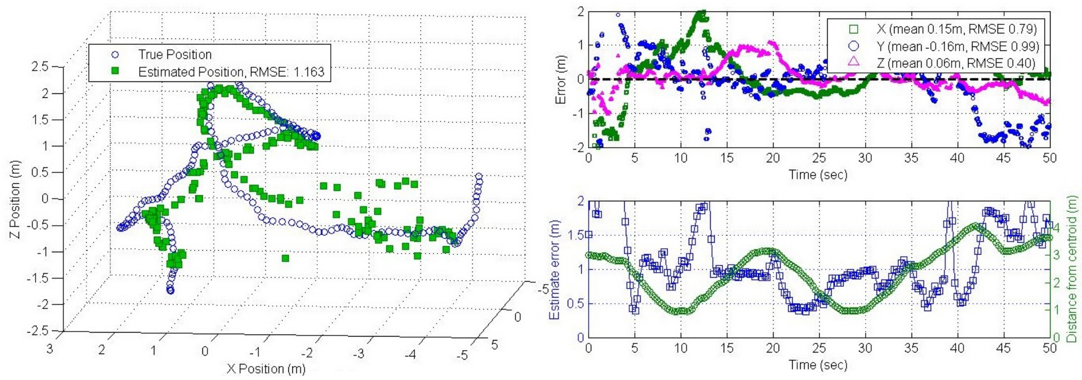


Figure 2.8: Localization errors for DOPT in 3D for a single mobile node. Spatial errors (left) are shown with corresponding per-axis errors by time (top right). Additionally, the error is plotted against the mobile nodes distance from the network centroid (bottom right).

2.5 Conclusion

In this chapter we described how some problems related to localization and clock parameter estimation can be cast as maximum likelihood estimation problems that are based in optimization. We then presented a distributed algorithm that iteratively computes the optimal solution to these constrained optimizations, using only locally available measurements.

Sufficient conditions for local asymptotic stability were derived by linearizing the system that described the evolution of the algorithm around an optimal solution. The resulting conditions are matrix inequalities that depend on the system constraints and on how the nodes are connected in the network.

A range-based localization problem was used to illustrate the proposed algorithm in simulation and we showed that the nodes' estimated positions converge to their true position in the presence of noise and with random initial estimates.

Finally, experiments using custom ultra-wideband radio devices were performed for a combined localization and clock synchronization problem with pseudo-range measurements with clock drifts and biases. The results showed localization error of 30 cm and synchronization error of 3 micro-seconds. On average, the proposed method outperformed three other standard distributed methods found in the literature for static localization.

Chapter 3

Distributed Coordination for Multi-Agent Systems

Parts of this chapter come from [18] and [19].

In this chapter we consider the problem of coordination of multi-agent systems. We propose a centralized model predictive control based solution, and explore the communication between agents to design two iterative algorithms that distributively solve the optimization problem. These algorithms differ in the range in which they need to communicate to obtain the data for local computations. When the agents communicate with their 2-hop neighbors, we show that the algorithm converges to the solution of the centralized model predictive control problem. Practical conditions for termination of the algorithm are given and they ensure the coordination of the agents. If the communication range is constrained to the 1-hop neighbors, the algorithm in general converges to a suboptimal solution of the original problem. Nevertheless, this optimality gap is small and in some cases zero.

Structure This chapter is organized as follows. In Section 3.1 we formulate the leader-follower synchronization problem and a centralized model predictive control based solution is presented in Section 3.2. Two distributed approaches that estimate the centralized solution are proposed and analyzed in Section 3.3, one that requires 2-hop neighbor information, and another that utilizes only 1-hop neighbor information. The algorithms are illustrated in simulation for an unstable linear system in Section 3.4 and concluding remarks are given in Section 3.5

3.1 Problem Formulation

Consider the following linear discrete-time multi-agent dynamical system

$$x_{t+1}^0 = Ax_t^0, \qquad x_{t+1}^i = Ax_t^i + Bu_t^i, \qquad i = \{1, 2, \ldots, N\}, \quad t \geqslant 0, \tag{3.1}$$

where $x_t^0 \in \mathbb{R}^{n_x}$ is the state of the leader agent, $x_t^i \in \mathbb{R}^{n_x}$ is the state of the ith follower agent, and $u_t^i \in \mathbb{R}^{n_u}$ its control input at time t. All agents have identical dynamics defined by the matrices $A \in \mathbb{R}^{n_x \times n_x}$ and $B \in \mathbb{R}^{n_x \times n_u}$.

The follower agents are able to exchange their state and control input via pair-wise communication restricted by an undirected graph $\mathcal{G} = (\mathcal{N}, \mathcal{E})$, where $\mathcal{N} = \{1, 2, \ldots, N\}$

is the set of follower agents and $\mathcal{E} \subseteq \mathcal{N} \times \mathcal{N}$ is the set of edges.

We are interested in designing local control inputs u_t^i to achieve synchronization with the leader. Mathematically this is defined as

$$\lim_{t\to\infty} \|x_t^i - x_t^0\| = 0, \qquad \forall i \in \mathcal{N}.$$

To study the synchronization problem, we start by defining the local tracking error as

$$\varepsilon_t^i \coloneqq \sum_{j\in\mathcal{N}_i} (x_t^j - x_t^i) + g^i(x_t^0 - x_t^i), \qquad \forall i \in \mathcal{N}, \quad t \geqslant 0,$$

where $\mathcal{N}_i \subset \mathcal{N}$ is the set of neighbors of agent i. A subset of the follower agents has access to the leader agent state and we represent that by $g^i = 1$ if agent i has access to the leader's state and $g^i = 0$ otherwise. The dynamics of this local error can be written as

$$\varepsilon_{t+1}^i = A\varepsilon_t^i - (g^i + d^i)Bu_t^i + B\sum_{j\in\mathcal{N}_i} u_t^j, \qquad \forall i \in \mathcal{N}, \tag{3.2}$$

where $d^i \coloneqq |\mathcal{N}_i|$ is the degree of agent i.

The following assumptions are standard and will be used later to establish synchronization.

Assumption 1 (Local controllability)**.** The pair (A, B) is controllable. □

Assumption 2 (Leader-follower connectivity)**.** Each connected component of $\mathcal{G}$ is connected to the leader. $\square$

Remark 3. Note that we do not require for the graph $\mathcal{G}$ to be connected. However, Assumption 2 is equivalent to the overall graph consisting of the union of the followers graph $\mathcal{G}$ with the graph spanned by the leader to be connected. $\square$

Definition 1 (Augmented Laplacian)**.** For a graph $\mathcal{G}$ we define the Augmented Laplacian matrix of a graph as

$$\bar{L} \coloneqq G + L,$$

where $L \coloneqq D - A_d \in \mathbb{R}^{N\times N}$ is the Laplacian matrix of $\mathcal{G}$, $D \coloneqq \operatorname{diag}(d^1, d^2, \dots, d^N) \in \mathbb{R}^{N\times N}$ is the degree matrix, $G \coloneqq \operatorname{diag}(g^1, g^2, \dots, g^N) \in \mathbb{R}^{N\times N}$ the matrix representing the coupling with the leader agent, and $A_d \in \mathbb{R}^{N\times N}$ the adjacency matrix. $\square$

The next result follows from the connectivity condition in Assumption 2.

Proposition 1. *Consider the graph $\mathcal{G}$ and let Assumption 2 hold. Then the Augmented Laplacian $\bar{L}$ is positive definite and has full rank.* $\square$

Proof. The Laplacian L of a graph with k connected components can be rearranged as

$$P^\top L P = \begin{bmatrix} L_1 & & \\ & \ddots & \\ & & L_k \end{bmatrix},$$

where $P \in \mathbb{R}^{N\times N}$ is a permutation matrix, and L_i corresponds to the Laplacian of a subgraph formed from the ith connected component. Similarly, G can be rearranged as

$$P^\top G P = \begin{bmatrix} G_1 & & \\ & \ddots & \\ & & G_k \end{bmatrix},$$

where G_i is diagonal with 0s or 1s in the diagonal. Since L_i is the Laplacian of the ith connected component, then $L_i \geqslant 0$ and for any vector v_i of a appropriate dimension such that $v_i \neq 0$, $v_i^\top L_i v_i = 0$ if and only if $v_i = \alpha\mathbf{1}$, where $\alpha \in \mathbb{R}$ and $\mathbf{1}$ is the vector of 1s. From Assumption 2 every G_i block has at least one diagonal entry equal to 1, thus $G_i \geqslant 0$ and $v_i^\top G_i v_i^\top > 0$ for any nonzero v_i in the kernel of L_i. Hence for any v_i,

$$v_i^\top (G_i + L_i) v_i > 0. \tag{3.3}$$

and we conclude that $G_i + L_i > 0$ for $i = 1, 2, \ldots, k$. Hence,

$$\bar{L} = P^\top \begin{bmatrix} G_1 + L_1 & & \\ & \ddots & \\ & & G_k + L_k \end{bmatrix} P,$$

is positive definite and $\operatorname{rank}(\bar{L}) = N$. ■

We evaluate the overall system controllability by considering the combined dynamics of the follower agents. For that, we define the multi-agent state and control at time t by $\varepsilon_t \coloneqq (\varepsilon_t^1, \varepsilon_t^2, \ldots, \varepsilon_t^N) \in \mathbb{R}^{Nn_x}$ and $u_t \coloneqq (u_t^1, u_t^2, \ldots, u_t^N) \in \mathbb{R}^{Nn_u}$, respectively. Then, the overall error dynamics derived from (3.2) is

$$\varepsilon_{t+1} = \bar{A}\varepsilon_t + \bar{B}u_t, \tag{3.4}$$

where $\bar{A} \coloneqq I_N \otimes A = \mathrm{diag}(A, \ldots, A) \in \mathbb{R}^{Nn_x \times Nn_x}$ and $\bar{B} \coloneqq -\bar{L} \otimes B \in \mathbb{R}^{Nn_x \times Nn_u}$.

The following corollary establishes the controllability of the multi-agent system.

Corollary 1 (Controllability of the multi-agent system)**.** *Consider the system in* (3.4) *and let Assumption 1 and 2 hold. Then the pair* $(\bar{A}, \bar{B})$ *is controllable.* □

Proof. Let $\bar{\mathcal{C}} \coloneqq \begin{bmatrix} \bar{B} & \bar{A}\bar{B} & \ldots \bar{A}^{Nn_x-1}\bar{B} \end{bmatrix}$ be the controllability matrix for the system in (3.4). Then,

$$\begin{aligned}
\bar{\mathcal{C}} &= \begin{bmatrix} \bar{L} \otimes B & (I_N \otimes A)(\bar{L} \otimes B) & \cdots & (I_N \otimes A^{Nnx-1})(\bar{L} \otimes B) \end{bmatrix} \\
&= \begin{bmatrix} \bar{L} \otimes B & \bar{L} \otimes AB & \cdots & \bar{L} \otimes A^{Nn_x-1}B \end{bmatrix} \\
&= \begin{bmatrix}
\bar{l}_{11}B & \cdots & \bar{l}_{1N}B & \bar{l}_{11}AB & \cdots & \bar{l}_{1N}AB & \cdots & \bar{l}_{11}A^{Nn_x-1}B & \cdots & \bar{l}_{1N}A^{Nn_x-1}B \\
\bar{l}_{21}B & \cdots & \bar{l}_{2N}B & \bar{l}_{21}AB & \cdots & \bar{l}_{2N}AB & \cdots & \bar{l}_{21}A^{Nn_x-1}B & \cdots & \bar{l}_{2N}A^{Nn_x-1}B \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
\bar{l}_{N1}B & \cdots & \bar{l}_{NN}B & \bar{l}_{N1}AB & \cdots & \bar{l}_{NN}AB & \cdots & \bar{l}_{N1}A^{Nn_x-1}B & \cdots & \bar{l}_{NN}A^{Nn_x-1}B
\end{bmatrix},
\end{aligned}$$

where

$$\bar{L} = \begin{bmatrix} \bar{l}_{11} & \cdots & \bar{l}_{1N} \\ \bar{l}_{21} & \cdots & \bar{l}_{2N} \\ \vdots & \cdots & \vdots \\ \bar{l}_{N1} & \cdots & \bar{l}_{NN} \end{bmatrix}.$$

Since the permutation of columns does not alter the rank of a matrix, we have that

$$\begin{aligned}
\mathrm{rank}(\bar{\mathcal{C}}) &= \mathrm{rank}\left(\begin{bmatrix} \bar{l}_{11}B & \cdots & \bar{l}_{11}A^{Nn_x-1}B & \bar{l}_{12}B & \cdots & \bar{l}_{12}A^{Nn_x-1}B & \cdots & \bar{l}_{1N}B & \cdots & \bar{l}_{1N}A^{Nn_x-1}B \\ \bar{l}_{21}B & \cdots & \bar{l}_{21}A^{Nn_x-1}B & \bar{l}_{22}B & \cdots & \bar{l}_{22}A^{Nn_x-1}B & \cdots & \bar{l}_{2N}B & \cdots & \bar{l}_{2N}A^{Nn_x-1}B \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \bar{l}_{N1}B & \cdots & \bar{l}_{N1}A^{Nn_x-1}B & \bar{l}_{N2}B & \cdots & \bar{l}_{N2}A^{Nn_x-1}B & \cdots & \bar{l}_{NN}B & \cdots & \bar{l}_{NN}A^{Nn_x-1}B \end{bmatrix}\right) \\
&= \mathrm{rank}\left(\begin{bmatrix} \bar{l}_{11}\mathcal{C} & \bar{l}_{12}\mathcal{C} & \cdots & \bar{l}_{1N}\mathcal{C} \\ \bar{l}_{21}\mathcal{C} & \bar{l}_{22}\mathcal{C} & \cdots & \bar{l}_{2N}\mathcal{C} \\ \vdots & \vdots & \vdots & \vdots \\ \bar{l}_{N1}\mathcal{C} & \bar{l}_{N2}\mathcal{C} & \cdots & \bar{l}_{NN}\mathcal{C} \end{bmatrix}\right) \\
&= \mathrm{rank}(\bar{L} \otimes \mathcal{C})
\end{aligned}$$

$$= \text{rank}(\bar{L})\,\text{rank}\,(\mathcal{C})\,,$$

where $\mathcal{C} \coloneqq \begin{bmatrix} B & AB & \dots & A^{Nnx-1}B \end{bmatrix}$. Assumption 1 implies that $\text{rank}(\mathcal{C}) = n_x$ and from Proposition 1 we have that $\text{rank}(\bar{L}) = N$. We conclude that $\text{rank}(\bar{\mathcal{C}}) = Nn_x$ and the pair $(\bar{A}, \bar{B})$ is controllable. ■

3.2 Centralized Multi-Agent Model Predictice Control

As discussed before, our goal is to compute the local control inputs at time t, $u_t^i \in \mathbb{R}^{n_u}$, $\forall i \in \mathcal{N}$ that achieve synchronization. For that consider the following optimization problem

$$\min_{u_{t:t+T-1}} J(u_{t:t+T-1}; \varepsilon_t), \tag{3.5}$$

where the cost is defined by

$$J(u_{t:t+T-1}; \varepsilon_t) \coloneqq \frac{1}{2} \sum_{i \in \mathcal{N}} \left[\sum_{k=t}^{t+T-1} ({\varepsilon_k^i}^\top Q \varepsilon_k^i + {u_k^i}^\top R u_k^i) + {\varepsilon_{t+T}^i}^\top Q_f \varepsilon_{t+T}^i \right],$$

where $T \in \mathbb{Z}_{>0}$ is the horizon length and $Q \in \mathbb{R}^{n_x \times n_x}$, $R \in \mathbb{R}^{n_u \times n_u}$, $Q_f \in \mathbb{R}^{n_x \times n_x}$, are the positive definite weighting matrices. This cost can be further represented as

$$J(u_{t:t+T-1}; \varepsilon_t) = \frac{1}{2} \sum_{k=t}^{t+T-1} (\varepsilon_k^\top \bar{Q} \varepsilon + u_k^\top \bar{R} u_k) + \frac{1}{2} \varepsilon_{t+T}^\top \bar{Q}_f \varepsilon_{t+T}, \tag{3.6}$$

where $\bar{Q} \coloneqq \mathrm{diag}(Q, Q, \ldots, Q) \in \mathbb{R}^{Nn_x \times Nn_x}$, $\bar{R} \coloneqq \mathrm{diag}(R, R, \ldots, R) \in \mathbb{R}^{Nn_u \times Nn_u}$, and $\bar{Q}_f \coloneqq \mathrm{diag}(Q_f, Q_f, \ldots, Q_f) \in \mathbb{R}^{Nn_x \times Nn_x}$ are the overall weighting matrices obtained by stacking the follower agents weighting matrices.

The optimization variable is the vector of length T defined by $u_{t:t+T-1} \coloneqq (u_t, u_{t+1}, \ldots, u_{t+T-1}) \in \mathbb{R}^{NTn_u}$, where $u_t \coloneqq (u_t^1, u_t^2, \ldots, u_t^N) \in \mathbb{R}^{Nn_x}$ contains the optimization variables for every follower agent at time t.

For the Centralized Multi-Agent Model Predictive Control (CMAMPC) problem, at each time we solve (3.5) subject to the dynamics constraint in (3.4) to obtain the vector $u^*_{t:t+T-1}$ that contains the optimal control input sequence of length T for every follower agent. From this optimal control sequence, only the element corresponding to the control input at time t, u_t^* is applied to (3.4) and the process is repeated at the next time step. The closed-loop system obtained by utilizing the CMAMPC is

$$\varepsilon_{t+1} = \bar{A}\varepsilon_t + \bar{B}u_t^*. \tag{3.7}$$

The next result is inspired by MPC stability results [40] and establishes a sufficient condition on the terminal cost that ensures asymptotic stability.

Theorem 2 (Asymptotic stability of the CMAMPC)**.** *Consider the system in* (3.7) *and suppose Assumption 2 hold. Let the terminal cost weighting matrix in* (3.6) *be* $\bar{Q}_f = I \otimes P$*, with* $P > 0$ *such that there exists a matrix* K *for which*

$$(A - BK)^\top P(A - BK) - P + Q + \theta K^\top RK = 0, \tag{3.8}$$

for some scalar θ *such that*

$$\theta \geqslant \bar{\lambda}\big(\bar{L}^{-2}\big). \tag{3.9}$$

Then the origin is globally asymptotically stable. □

Proof. For an unconstrained MPC problem with finite horizon quadratic cost of the type of (3.5), we have that if $\bar{Q}_f$ is such that

$$(\bar{A} + \bar{B}\bar{K})^\top \bar{Q}_f(\bar{A} + \bar{B}\bar{K}) + \bar{Q} + \bar{K}^\top \bar{R}\bar{K} - \bar{Q}_f \leqslant 0, \tag{3.10}$$

then the origin of (3.7) is asymptotically stable [40]. In Proposition 1 we showed that under Assumption 2 the Augmented Laplacian $\bar{L}$ of a graph is full rank therefore invertible. Letting $\bar{K} = \bar{L}^{-1} \otimes K$ and $\bar{Q}_f = I_N \otimes P$, and utilizing the definitions for $\bar{A}$, $\bar{B}$, $\bar{Q}$, and $\bar{R}$, the left-hand side of (3.10) can be expanded as

$$\big(I_N \otimes A - (\bar{L} \otimes B)(\bar{L}^{-1} \otimes K)\big)^\top (I_N \otimes P)\big(I_N \otimes A - (\bar{L} \otimes B)(\bar{L}^{-1} \otimes K)\big)$$

$$+ I_N \otimes Q + (\bar{L}^{-1} \otimes K)^\top (I_N \otimes R)(\bar{L}^{-1} \otimes K) - I_N \otimes P$$

$$= I_N \otimes \big((A - BK)^\top P(A - BK) - P + Q + \theta K^\top RK\big) + (\bar{L}^{-2} - \theta I_N) \otimes (K^\top RK).$$

From the condition in (3.8) we have that (3.10) holds completing the proof. ■

Remark 4. For (3.8) to hold the pair (A, B) must be stabilizable. Under the controllability contidion in Assumption 1 we have that there exists a matrix K such that $A - BK$ is Schur and there exists a unique $P > 0$ that is the solution to (3.8). □

Remark 5. In MPC applications, the terminal cost defined by the matrix $\bar{Q}_f$ could be omitted when the horizon is sufficiently long [22], although this could introduce challenges in terms of computation, as the size of the problem grows with the length of the horizon.

□

Remark 6. Note that (3.9) implies that the choice of the local parameter Q_f is conditioned on the knowledge of the graph topology (since it depends on $\bar{L}^{-2}$). However, this can be leveraged by utilizing results on the upper and lower bounds of the eigenvalue of the Laplacian matrix [1]. □

Theorem 2 provides a guideline on the selection of the terminal weighting matrix $\bar{Q}_f$ that ensures closed-loop stability. Forcing $\bar{Q}_f$ to be a block diagonal solution to (3.10) although restrictive at first sight, will be shown later to be suitable for the design of distributed algorithms to solve the optimization in (3.5).

The cost function in (3.6) is of the form $J(\cdot) = \sum_{i \in \mathcal{N}} J^i(\cdot)$, the sum of local quadratic costs that can be expressed alternatively as

$$J^i(u^i_{t:t+T-1};\varepsilon^i_t) \coloneqq \frac{1}{2}\sum_{k=t}^{t+T-1}({\varepsilon^i_k}^\top Q\varepsilon^i_k + {u^i_k}^\top R u^i_k) + \frac{1}{2}{\varepsilon^i_{t+T}}^\top Q_f \varepsilon^i_{t+T}$$
$$= \frac{1}{2}({\varepsilon^i_{t+1:t+T}}^\top \mathcal{Q}\varepsilon^i_{t+1:t+T} + {u^i_{t:t+T-1}}^\top \mathcal{R} u^i_{t:t+T-1} + {\varepsilon^i_t}^\top Q\varepsilon^i_t), \tag{3.11}$$

where $\mathcal{Q} \coloneqq \mathrm{diag}(Q,\ldots,Q,Q_f) \in \mathbb{R}^{Tn_x\times Tn_x}$, $\mathcal{R} \coloneqq \mathrm{diag}(R,R,\ldots,R) \in \mathbb{R}^{Tn_u\times Tn_u}$. The vectors $\varepsilon^i_{t+1:t+T} \coloneqq (\varepsilon^i_{t+1},\varepsilon^i_{t+1},\ldots,\varepsilon^i_{t+T}) \in \mathbb{R}^{Tn_x}$ and $u^i_{t:t+T-1} \coloneqq (u^i_t,u^i_{t+1},\ldots,u^i_{t+T-1}) \in \mathbb{R}^{Tn_u}$ represent, respectively, the sequence of length T of state and control input values for the ith agent. Then, the optimization in (3.5) is equivalent to

$$\min_{u_{t:t+T-1}} \frac{1}{2}\sum_{i\in\mathcal{N}} {\varepsilon^i_{t+1:t+T}}^\top \mathcal{Q}\varepsilon^i_{t+1:t+T} + {u^i_{t:t+T-1}}^\top \mathcal{R} u^i_{t:t+T-1} + {\varepsilon^i_t}^\top Q\varepsilon^i_t, \tag{3.12}$$

where the optimization variable is $u_{t:t+T-1} \coloneqq (u^1_{t:t+T-1},u^2_{t:t+T-1},\ldots,u^N_{t:t+T-1})$ and the problem is to be solved subject to the dynamic constraint in (3.2) for all agents.

Lemma 3 (Closed form expression for the CMAMPC solution)**.** *Consider the optimization in* (3.12) *and suppose that Assumption 2 holds. Then the unique global minimum* $u^*_{t:t+T-1} \coloneqq (u^{1*}_{t:t+T-1},u^{2*}_{t:t+T-1},\ldots,u^{N*}_{t:t+T-1})$ *is given by*

$$u^*_{t:t+T-1} = (\bar{L}^2\otimes(\mathcal{B}^\top\mathcal{Q}\mathcal{B}) + I_N\otimes\mathcal{R})^{-1}(\bar{L}\otimes(\mathcal{B}^\top\mathcal{Q}\mathcal{A}))\varepsilon_t, \tag{3.13}$$

and the value of the minimum is

$$J(u^*_{t:t+T-1}; \varepsilon_t) = \frac{1}{2}\varepsilon_t^\top \Psi \varepsilon_t, \tag{3.14}$$

where

$$\Psi \coloneqq I_N \otimes (\mathcal{A}^\top \mathcal{Q}\mathcal{A} + Q) - \bar{L} \otimes (\mathcal{A}^\top \mathcal{Q}\mathcal{B})(\bar{L}^2 \otimes (\mathcal{B}^\top \mathcal{Q}\mathcal{B}) + I_N \otimes \mathcal{R})^{-1}(\bar{L} \otimes (\mathcal{B}^\top \mathcal{Q}\mathcal{A})), \tag{3.15}$$

and

$$\mathcal{A} \coloneqq \begin{bmatrix} A \\ A^2 \\ \vdots \\ A^T \end{bmatrix} \in \mathbb{R}^{Tn_x \times n_x}, \quad \mathcal{B} \coloneqq \begin{bmatrix} B & 0 & \cdots & 0 \\ AB & B & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ A^{T-1}B & A^{T-2}B & \cdots & B \end{bmatrix} \in \mathbb{R}^{Tn_x \times Tn_u}. \quad \square$$

Proof. The closed form expression for the optimal control input sequence can be obtained by computing the first-order optimality condition of the problem in (3.12). For that, we first write the dynamics for a sequence of tracking error of length T obtained from the local dynamics in (3.2)

$$\varepsilon^i_{t+1:t+T} = \mathcal{A}\varepsilon^i_t - (g^i + d^i)\mathcal{B}u^i_{t:t+T-1} + \mathcal{B}\sum_{j \in \mathcal{N}_i} u^j_{t:t+T-1}, \tag{3.16}$$

where $u_{t:t+T-1} \coloneqq (u^1_{t:t+T-1}, u^2_{t:t+T-1}, \ldots, u^N_{t:t+T-1}) \in \mathbb{R}^{TNn_u}$. Then, by incorporating

(3.16) into the cost function (3.11) we have that

$$
\begin{aligned}
J(u_{t:t+T-1};\varepsilon_t) = \ & \frac{1}{2}\sum_{i\in\mathcal{N}}\Bigg[\ {\varepsilon_t^i}^\top(\mathcal{A}^\top\mathcal{Q}\mathcal{A}+Q)\varepsilon_t^i + {u_{t:t+T-1}^i}^\top\big((g^i+d^i)^2\mathcal{B}^\top\mathcal{Q}\mathcal{B}+\mathcal{R}\big)u_{t:t+T-1}^i \\
& + \Bigg(\sum_{j\in\mathcal{N}_i}u_{t:t+T-1}^j\Bigg)^\top(\mathcal{B}^\top\mathcal{Q}\mathcal{B})\Bigg(\sum_{j\in\mathcal{N}_i}u_{t:t+T-1}^j\Bigg) - 2(g^i+d^i){\varepsilon_t^i}^\top(\mathcal{A}^\top\mathcal{Q}\mathcal{B})u_{t:t+T-1}^i \\
& - 2(g^i+d^i)\Bigg(\sum_{j\in\mathcal{N}_i}u_{t:t+T-1}^j\Bigg)^\top\mathcal{B}^\top\mathcal{Q}\mathcal{B}u_{t:t+T-1}^i + 2{\varepsilon_t^i}^\top\mathcal{A}^\top\mathcal{Q}\mathcal{B}\Bigg(\sum_{j\in\mathcal{N}_i}u_{t:t+T-1}^j\Bigg)\Bigg].
\end{aligned}
$$

By concatenation of the agents states and control inputs we obtain the compact form

$$
\begin{aligned}
J(u_{t:t+T-1};\varepsilon_t) = & \ \frac{1}{2}\varepsilon_t^\top\big(I_N\otimes(\mathcal{A}^\top\mathcal{Q}\mathcal{A}+Q)\big)\varepsilon_t - \varepsilon_t^\top\big(\bar{L}\otimes(\mathcal{A}^\top\mathcal{Q}\mathcal{B})\big)u_{t:t+T-1} \\
& + \frac{1}{2}u_{t:t+T-1}^\top\big(\bar{L}^2\otimes(\mathcal{B}^\top\mathcal{Q}\mathcal{B})+I_N\otimes\mathcal{R}\big)u_{t:t+T-1}. \qquad (3.17)
\end{aligned}
$$

The optimal control input sequence $u_{t:t+T-1}^*$ that solves (3.12) satisfies the following first-order optimality condition

$$
\nabla_{u_{t:t+T-1}}J(u_{t:t+T-1};\varepsilon_t)\big|_{u_{t:t+T-1}=u_{t:t+T-1}^*}=0,
$$

from which we obtain

$$
(\bar{L}^2\otimes(\mathcal{B}^\top\mathcal{Q}\mathcal{B})+I_N\otimes\mathcal{R})u_{t:t+T-1}^* - (\bar{L}\otimes(\mathcal{B}^\top\mathcal{Q}\mathcal{A}))\varepsilon_t = 0. \qquad (3.18)
$$

To prove uniqueness we show that $\bar{L}^2\otimes(\mathcal{B}^\top\mathcal{Q}\mathcal{B})+I_N\otimes\mathcal{R}$ is positive definite and therefore invertible. From Proposition 1 we have that $\bar{L} > 0$ and so is $\bar{L}^2$. Since $\mathcal{Q} > 0$ we have that $\mathcal{B}^\top \mathcal{Q} \mathcal{B} \geqslant 0$, and since $\mathcal{R} > 0$ we conclude that $\bar{L}^2 \otimes (\mathcal{B}^\top \mathcal{Q} \mathcal{B}) + I_N \otimes \mathcal{R} > 0$. Solving (3.18) for $u^*_{t:t+T-1}$ we obtain (3.13) and the minimum cost in (3.14) comes from a direct substitution of (3.13) in (3.17). ∎

As discussed before, for the CMAMPC problem only the first element of the computed optimal control sequence in (3.13) is fed back into the system, and in the next time instant a new sequence is recomputed. The control input utilized to drive the system at time t is

$$u^*_t = \mathcal{K}^* \varepsilon_t \tag{3.19}$$

where

$$\mathcal{K}^* \coloneqq \mathcal{V}(\bar{L}^2 \otimes (\mathcal{B}^\top \mathcal{Q} \mathcal{B}) + I_N \otimes \mathcal{R})^{-1}(\bar{L} \otimes (\mathcal{B}^\top \mathcal{Q} \mathcal{A})), \tag{3.20}$$

and $\mathcal{V} \in \mathbb{R}^{Nn_u \times TNn_u}$ is a matrix that selects the control input corresponding to time t from the sequence $u^*_{t:t+T-1}$ of length T. The closed-loop system obtained by combining (3.19) with (3.4) is the following linear system

$$\varepsilon_{t+1} = \bar{A}^* \varepsilon_t, \tag{3.21}$$

where $\bar{A}^* \coloneqq \bar{A} + \bar{B}\mathcal{K}^*$ is Schur stable in view of Theorem 2.

3.3 Distributed Solutions to the Centralized Multi-Agent Model Preditive Control

In this section we introduce the formulations and algorithms that will allow us to solve in a distributed fashion the optimization problem in (3.5) that constitutes the CMAMPC. By distributed we mean that the computations will be carried out at each agent utilizing, exclusively, the values of state and control input that are available from the communication with their neighbors.

3.3.1 The 2-hop Partial Optimization

For each $i \in \mathcal{N}$, consider the following optimization that is local to agent i

$$\min_{u^i_{t:t+T-1}} \bar{J}^i(u^i_{t:t+T-1}; \varepsilon^i_t, \varepsilon^{-i_1}_t, u^{-i_1}_{t:t+T-1}, u^{-i_2}_{t:t+T-1}), \tag{3.22}$$

where the local cost functions are of the form

$$\bar{J}^i(\cdot) \coloneqq J^i(\cdot) + \sum_{\ell \in \mathcal{N}_i} J^\ell(\cdot),$$

with function $J^i(\cdot)$ and $J^\ell(\cdot)$ defined as

$$J^i(u^i_{t:t+T-1}; \varepsilon^i_t) \coloneqq \frac{1}{2} \sum_{k=t}^{t+T-1} ({\varepsilon^i_k}^\top Q \varepsilon^i_k + {u^i_k}^\top R u^i_k) + \frac{1}{2} {\varepsilon^i_{t+T}}^\top Q_f \varepsilon^i_{t+T}$$

$$= \frac{1}{2}({\varepsilon^i_{t+1:t+T}}^\top \mathcal{Q} \varepsilon^i_{t+1:t+T} + {u^i_{t:t+T-1}}^\top \mathcal{R} u^i_{t:t+T-1} + {\varepsilon^i_t}^\top Q \varepsilon^i_t).$$

The optimization in (3.22) is subject to the dynamic constraint

$$\varepsilon^i_{k+1} = A\varepsilon^i_k - (g^i + d^i)Bu^i_k + B\sum_{j\in\mathcal{N}_i} u^j_k, \qquad \forall k \in \{t, t+1, \ldots, t+T-1\}, \tag{3.23}$$

for agent i and for every agent $\ell \in \mathcal{N}_i$.

The parameters to the optimization are: the local tracking error ε^i_t; the set of tracking errors associated with the 1-hop neighbors of i denoted by $\varepsilon^{-i_1}_t \coloneqq \{\varepsilon^\ell_t : \ell \in \mathcal{N}_i\}$; the control input sequence from the 1-hop neighbors $u^{-i_1}_{t:t+T-1}$; and the set of control input sequences from 2-hop neighbors denoted by $u^{-i_2}_{t:t+T-1} \coloneqq \{u^\ell_{t:t+T-1} : \ell \in \bigcup_{j\in\mathcal{N}_i}\mathcal{N}_j \backslash \{\mathcal{N}_i \cup i\}\}$.

It is important to note that $u^i_{t:t+T-1}$ is the optimization variable for (3.22), but the optimization criterion depends on the tracking error of the 1-hop neighbor and the control input sequences from 1-hop and 2-hop neighbors of i. This dependence arises from the fact that we constraint the local and the neighbor's state sequences to be a solution of (3.23). This observation will become evident below.

Denoting by $\bar{u}^{i*}_{t:t+T-1}$ the local optimal control input sequence that solves (3.22), the first-order optimality condition for this problem is

$$\nabla_{u^i_{t:t+T-1}} \bar{J}^i(u^i_{t:t+T-1}; \varepsilon^i_t, \varepsilon^{-i_1}_t, u^{-i_1}_{t:t+T-1}, u^{-i_2}_{t:t+T-1})\Big|_{u^i_{t:t+T-1} = \bar{u}^{i*}_{t:t+T-1}} = 0,$$

that results in

$$\big((g^i + d^i)^2 \mathcal{B}^\top \mathcal{Q}\mathcal{B} + \mathcal{R}\big)\bar{u}^{i*}_{t:t+T-1} - (g^i + d^i)\mathcal{B}^\top \mathcal{Q}\mathcal{B} \sum_{j\in\mathcal{N}_i} u^j_{t:t+T-1} - (g^i + d^i)(\mathcal{B}^\top \mathcal{Q}\mathcal{A})\varepsilon^i_t \\ + \sum_{\ell\in\mathcal{N}_i} \left[\mathcal{B}^\top \mathcal{Q}\mathcal{B} \sum_{j\in\mathcal{N}_\ell} u^j_{t:t+T-1} - (g^\ell + d^\ell)\mathcal{B}^\top \mathcal{Q}\mathcal{B} u^\ell_{t:t+T-1} + \mathcal{B}^\top \mathcal{Q}\mathcal{A}\varepsilon^\ell_t\right] = 0. \quad (3.24)$$

All the conditions (3.24), one for each agent $i \in \mathcal{N}$, can be expressed in vector form as

$$\nabla_{u_{t:t+T-1}} \bar{J}(u_{t:t+T-1}; \varepsilon_t)\big|_{u_{t:t+T-1}=\bar{u}^*_{t:t+T-1}} \\ := \left[\nabla_{u^i_{t:t+T-1}} \bar{J}^i(u^i_{t:t+T-1}; \varepsilon^i_t, \varepsilon^{-i_1}_t, u^{-i_1}_{t:t+T-1}, u^{-i_2}_{t:t+T-1})\right]\Bigg|_{\substack{u_{t:t+T-1} \\ =\bar{u}^*_{t:t+T-1}}} = 0$$

for which we obtain

$$(\bar{L}^2 \otimes (\mathcal{B}^\top \mathcal{Q}\mathcal{B}) + I_N \otimes \mathcal{R})\bar{u}^*_{t:t+T-1} - (\bar{L} \otimes (\mathcal{B}^\top \mathcal{Q}\mathcal{A}))\varepsilon_t = 0. \quad (3.25)$$

Comparing (3.25) with (3.18), we conclude that if all agents simultaneously solve the optimization in (3.22) they will be able to find the global optimum to (3.5).

3.3.2 The 2HopDMAMPC Algorithm

For our first algorithm, we assume that each agent is able to store estimates of each of its neighbors' optimal solutions and re-transmit them to their 1-hop neighbors. In practice, this means that each agent has available the delayed values of the control sequence of their 1-hop and 2-hop neighbors and could therefore solve (3.22) based on these estimates.

This leads to the iterative 2-Hop Distributed Multi-Agent Model Predictive Control (2HopDMAMPC) Algorithm where $\hat{\hat{u}}^i_{t:t+T-1}(m)$ should be viewed as an estimate of $\bar{u}^{i*}_{t:t+T-1}$, at iteration m of the algorithm.

Algorithm 2 2HopDMAMPC Algorithm for agent i

Require: a tolerance $\delta > 0$ and local state ε^i_t

Broadcast ε^i_t to neighbors

$m \leftarrow 0$

Initialize $\hat{\hat{u}}^i_{t:t+T-1}(m)$

Broadcast $\hat{\hat{u}}^i_{t:t+T-1}(m)$ to neighbors

$m \leftarrow m + 1$

Initialize $\hat{\hat{u}}^i_{t:t+T-1}(m)$

repeat

  Broadcast $\hat{\hat{u}}^i_{t:t+T-1}(m)$ to neighbors

  Broadcast $\hat{\hat{u}}^j_{t:t+T-1}(m-1)$ $\forall j \in \mathcal{N}_i$ to neighbors

  $\hat{\hat{u}}^i_{t:t+T-1}(m+1) \leftarrow \arg\min_{u^i_{t:t+T-1}} \bar{J}^i\big(u^i_{t:t+T-1}; \varepsilon^i_t, \varepsilon^{-i_1}_t, \hat{\hat{u}}^{-i_1}_{t:t+T-1}(m), \hat{\hat{u}}^{-i_2}_{t:t+T-1}(m-1)\big)$

  $m \leftarrow m + 1$

  $error \leftarrow \left\| \begin{bmatrix} \hat{\hat{u}}^i_{t:t+T-1}(m-1) \\ \hat{\hat{u}}^i_{t:t+T-1}(m) \end{bmatrix} - \begin{bmatrix} \hat{\hat{u}}^i_{t:t+T-1}(m-2) \\ \hat{\hat{u}}^i_{t:t+T-1}(m-1) \end{bmatrix} \right\|$

until $error \leqslant \delta \|\varepsilon^i_t\|$

  return $\hat{\hat{u}}^i_{t:t+T-1}(m)$

To study the convergence of the estimates obtained from the 2HopDMAMPC Algorithm we treat the evolution of the estimates, $\hat{\hat{u}}^i_{t:t+T-1}(m)$, $\forall i \in \mathcal{N}$ as a dynamical system. The optimization solved at line 10 of the algorithm satisfies the following first-order optimality condition $\forall i \in \mathcal{N}$

$$\nabla_{u^i_{t:t+T-1}} \bar{J}^i(u^i_{t:t+T-1}; \varepsilon^i_t, \varepsilon^{-i_1}_t, \hat{\hat{u}}^{-i_1}_{t:t+T-1}(m+1), \hat{\hat{u}}^{-i_2}_{t:t+T-1}(m))\Big|_{u^i_{t:t+T-1}=\hat{\hat{u}}^i_{t:t+T-1}(m+2)} = 0,$$

which is

$$\begin{aligned}\big((g^i + d^i)^2\mathcal{B}^\top\mathcal{Q}\mathcal{B} + \mathcal{R}\big)\hat{\hat{u}}^i_{t:t+T-1}(m+2) - (g^i + d^i)\mathcal{B}^\top\mathcal{Q}\mathcal{B}\sum_{j\in\mathcal{N}_i}\hat{\hat{u}}^j_{t:t+T-1}(m+1) \\ - (g^i + d^i)(\mathcal{B}^\top\mathcal{Q}\mathcal{A})\varepsilon^i_t + \sum_{\ell\in\mathcal{N}_i}\Bigg[(\mathcal{B}^\top\mathcal{Q}\mathcal{B})\sum_{j\in\mathcal{N}_\ell}\hat{\hat{u}}^j_{t:t+T-1}(m) \\ - (g^\ell + d^\ell)\mathcal{B}^\top\mathcal{Q}\mathcal{B}\hat{\hat{u}}^\ell_{t:t+T-1}(m+1) + \mathcal{B}^\top\mathcal{Q}\mathcal{A}\varepsilon^\ell_t\Bigg] = 0. \qquad (3.26)\end{aligned}$$

Denoting by $\hat{\hat{u}}_{t:t+T-1}(m) \coloneqq \big(\hat{\hat{u}}^1_{t:t+T-1}(m), \hat{\hat{u}}^2_{t:t+T-1}(m), \dots, \hat{\hat{u}}^N_{t:t+T-1}(m)\big)$ the collection of estimates for all the agents, then the ensemble of first-order optimality condition (3.26) can be written as

$$\bar{\mathcal{M}}_1\hat{\hat{u}}_{t:t+T-1}(m+2) - \bar{\mathcal{M}}_2\hat{\hat{u}}_{t:t+T-1}(m+1) + \bar{\mathcal{M}}_3\hat{\hat{u}}_{t:t+T-1}(m) - \bar{\mathcal{O}}\varepsilon_t = 0. \qquad (3.27)$$

where the system matrices are $\bar{\mathcal{M}}_1 \coloneqq (G+D)^2 \otimes (\mathcal{B}^\top\mathcal{Q}\mathcal{B}) + I_N \otimes \mathcal{R}$, $\bar{\mathcal{M}}_2 \coloneqq (A_d(G+D) + (G+D)A_d) \otimes (\mathcal{B}^\top\mathcal{Q}\mathcal{B})$, $\bar{\mathcal{M}}_3 \coloneqq A_d^2 \otimes (\mathcal{B}^\top\mathcal{Q}\mathcal{B})$, and $\bar{\mathcal{O}} \coloneqq \bar{L} \otimes (\mathcal{B}^\top\mathcal{Q}\mathcal{A})$. Defining the augmented state by $w_{t:t+T-1}(m) \coloneqq \begin{bmatrix}\hat{\hat{u}}_{t:t+T-1}(m-1)^\top & \hat{\hat{u}}_{t:t+T-1}(m)^\top\end{bmatrix}^\top \in \mathbb{R}^{2NTn_u}$, we can express the second order dynamical system in (3.27) as the following first-order dynamical system with $\hat{\hat{u}}_{t:t+T-1}(m)$ as output by

$$\begin{aligned} w_{t:t+T-1}(m+1) &= \bar{\mathcal{M}} w_{t:t+T-1}(m) + \bar{\mathcal{P}} \varepsilon_t \\ \hat{\hat{u}}_{t:t+T-1}(m) &= \bar{\mathcal{C}} w_{t:t+T-1}(m), \end{aligned} \tag{3.28}$$

where

$$\bar{\mathcal{M}} \coloneqq \begin{bmatrix} 0 & I \\ -\bar{\mathcal{M}}_1^{-1}\bar{\mathcal{M}}_3 & \bar{\mathcal{M}}_1^{-1}\bar{\mathcal{M}}_2 \end{bmatrix} \in \mathbb{R}^{2NTn_u \times 2NTn_u} \qquad \bar{\mathcal{P}} \coloneqq \begin{bmatrix} 0 \\ \bar{\mathcal{M}}_1^{-1}\bar{\mathcal{O}} \end{bmatrix} \in \mathbb{R}^{2NTn_u \times Nn_x}$$
$$\bar{\mathcal{C}} \coloneqq \begin{bmatrix} 0_{NTn_u} & I_{NTn_u} \end{bmatrix} \in \mathbb{R}^{NTn_u \times 2NTn_u}.$$

The subscript in $w_{t:t+T-1}(m)$ denotes that the the evolution of iterations m of the system in (3.28) occurs at time instant t where a sequence of length T of control inputs is being estimated.

Theorem 3 (Convergence of the 2HopDMAMPC Algorithm to the CMAMPC solution)**.** *Consider the system in* (3.28)*, let* $\mathcal{R} = rI_{Tn_u}$*, with* $r > 0$*, and suppose that Assumption 2 holds. Then the estimates* $\hat{\hat{u}}_{t:t+T-1}$ *converge to the optimal solution in* (3.13)*, i.e.,* $\lim_{m\to\infty} \hat{\hat{u}}_{t:t+T-1}(m) = u^*_{t:t+T-1}$ *and the algorithm terminates in finite time.* □

Proof. The equilibrium output $\hat{\hat{u}}^*_{t:t+T-1} \coloneqq \lim_{m\to\infty} \hat{\hat{u}}_{t:t+T-1}(m)$ of the system in (3.28) satisfies

$$(\bar{L}^2 \otimes (\mathcal{B}^\top \mathcal{Q}\mathcal{B}) + I_N \otimes \mathcal{R})\hat{\hat{u}}^*_{t:t+T-1} - (\bar{L} \otimes (\mathcal{B}^\top \mathcal{Q}\mathcal{A}))\varepsilon_t = 0.$$

In Lemma 3 we showed that $\bar{L}^2 \otimes (\mathcal{B}^\top \mathcal{Q}\mathcal{B}) + I_N \otimes \mathcal{R}$ is invertible, hence the unique equilibrium $\hat{u}^*_{t:t+T-1}$ is equal to the optimal solution $u^*_{t:t+T-1}$ in (3.13) for the CMAMPC problem.

To show convergence let $v := \begin{bmatrix} v_1^\top & v_2^\top \end{bmatrix}^\top$ and λ be an arbitrary eigenvector and eigenvalue pair of the matrix $\bar{\mathcal{M}}$ in (3.28). Then,

$$\begin{bmatrix} 0_n & I_n \\ -\bar{\mathcal{M}}_1^{-1}\bar{\mathcal{M}}_3 & \bar{\mathcal{M}}_1^{-1}\bar{\mathcal{M}}_2 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} = \lambda \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} \Rightarrow \Omega(\lambda) v_1 = 0$$

where $\Omega(\lambda) := (\lambda(G+D) - A_d)^2 \otimes (\mathcal{B}^\top \mathcal{Q}\mathcal{B}) + \lambda^2 I_N \otimes \mathcal{R}$. The stability of $\bar{\mathcal{M}}$ is equivalent to $\Omega(\lambda)$ being nonsingular for any λ, such that $|\lambda| > 1$. The spectrum of $\Omega(\lambda)$, denoted by $\sigma(\Omega(\lambda))$, satisfies

$$\begin{aligned} \sigma(\Omega(\lambda)) &= \sigma\big((\lambda(G+D) - A_d)^2 \otimes (\mathcal{B}^\top \mathcal{Q}\mathcal{B}) + \lambda^2 I_N \otimes \mathcal{R}\big) \\ &= \big(\sigma(\lambda(G+D) - A_d)\big)^2 \sigma((\mathcal{B}^\top \mathcal{Q}\mathcal{B})) + \lambda^2 r, \end{aligned}$$

where we use the fact that $\mathcal{R} = rI$ and the eigenvalue properties when taking Kronecker products. Let η be an element of $\sigma(\lambda(G+D) - A_d)$, i.e., an arbitrary eigenvalue of $\lambda(G+D) - A_d$. Then from the Gersgorin disc theorem [25] we have that

$$\eta \in \bigcup_{i=1}^{N} \{z : |z - \lambda(g^i + d^i)| \leqslant d^i\}.$$

Hence, for an eigenvalue ω of $\Omega(\lambda)$ we have that

$$\begin{aligned} &\omega \in \bigcup_{i=1}^{N} \{z^2\beta + \lambda^2 r : |z - \lambda(g^i + d^i)| \leqslant d^i\} \quad \forall \beta \in \sigma(\mathcal{B}^\top \mathcal{Q}\mathcal{B})\backslash\{0\} \\ \Leftrightarrow \quad &\omega \in \bigcup_{i=1}^{N} \{y : |\sqrt{\frac{y}{\beta} - \lambda^2 \frac{r}{\beta}} - \lambda(g^i + d^i)| \leqslant d^i\} \quad \forall \beta \in \sigma(\mathcal{B}^\top \mathcal{Q}\mathcal{B})\backslash\{0\}, \end{aligned} \tag{3.29}$$

where β is a nonzero eigenvalue of $\mathcal{B}^\top \mathcal{Q}\mathcal{B}$. Let $\lambda = ce^{jw}$, where $c > 1$, and suppose that $0 \in \sigma(\Omega(\lambda))$. Then for all $i \in \mathcal{N}$ and all $\beta \in \sigma(\mathcal{B}^\top \mathcal{Q}\mathcal{B})\backslash\{0\}$ we have that

$$|\sqrt{\frac{rc^2}{\beta} e^{j(2w+\pi)}} - c(g^i + d^i)e^{jw}| \leqslant d^i \;\Rightarrow\; c\sqrt{\frac{r}{\beta} + (g^i + d^i)^2} \leqslant d^i,$$

which is a contradiction, since $\beta > 0$, and $r > 0$. We conclude that for any λ such that $|\lambda| > 1$ the matrix $\Omega(\lambda)$ is nonsingular, hence $\bar{\mathcal{M}}$ is Schur and the estimates $\hat{\hat{u}}_{t:t+T-1}$ asymptotically converge to $u^*_{t:t+T-1}$. ■

3.3.3 The 2-Hop Distributed Multi-Agent Model Predictive Control

As typical in MPC applications, only the first element from the estimated sequence $\hat{\hat{u}}_{t:t+T-1}$ is applied to the system at time t. The closed-loop system obtained by combining (3.4) with the control estimates computed from the 2HopDMAMPC Algorithm at iteration m is

$$\varepsilon_{t+1} = \bar{A}\varepsilon_t + \bar{B}\hat{\bar{u}}_t(m), \tag{3.30}$$

where

$$\hat{\bar{u}}_t(m) = \mathcal{V}\hat{\bar{u}}_{t:t+T-1}(m)$$

is the control corresponding to the action at time t. The matrix $\mathcal{V} \in \mathbb{R}^{Nn_u \times TNn_u}$ selects the first element from the sequence of inputs of length T. Since the control sequence estimate can be regarded as the output of the system in (3.28), then

$$\hat{\bar{u}}_t(m) = \mathcal{V}\bar{\mathcal{C}}w_{t:t+T-1}(m).$$

Given an initial condition $w_{t:t+T-1}(0)$, i.e., the initialization of the 2HopDMAMPC Algorithm in line 3 and 6, we have that the evolution in iterations of the estimates are

$$\hat{\bar{u}}_t(m) = \mathcal{V}\bar{\mathcal{C}}\bar{\mathcal{M}}^m w_{t:t+T-1}(0) + \bar{\mathcal{K}}(m)\varepsilon_t \tag{3.31}$$

where

$$\bar{\mathcal{K}}(m) \coloneqq \mathcal{V}\bar{\mathcal{C}} \sum_{\tau=0}^{m-1} \bar{\mathcal{M}}^{\tau}\bar{\mathcal{P}}. \tag{3.32}$$

From Theorem 3 we have that the estimates obtained from the distributed algorithm converge to the centralized control. Hence,

$$\lim_{m\to\infty} \hat{\hat{u}}_{t:t+T-1}(m) = u^*_{t:t+T-1} \Rightarrow \lim_{m\to\infty} \hat{\hat{u}}_t(m) = u^*_t = \mathcal{K}^*\varepsilon_t, \tag{3.33}$$

where we use the definition of the centralized optimal control in (3.19). Additionally, since $\bar{\mathcal{M}}$ is stable then the limit of (3.31) as the number of iterations approach infinity is

$$\lim_{m\to\infty} \hat{\hat{u}}_t(m) = \lim_{m\to\infty} \mathcal{V}\bar{\mathcal{C}}\bar{\mathcal{M}}^m w_{t:t+T-1}(0) + \bar{\mathcal{K}}(m)\varepsilon_t = \lim_{m\to\infty} \bar{\mathcal{K}}(m)\varepsilon_t. \tag{3.34}$$

From the definition of $\bar{\mathcal{K}}(m)$ in (3.32) and in light of (3.34) and (3.33) we conclude that

$$\lim_{m\to\infty} \bar{\mathcal{K}}(m) = \mathcal{V}\bar{\mathcal{C}} \sum_{\tau=0}^{\infty} \bar{\mathcal{M}}^\tau \bar{\mathcal{P}} = \mathcal{K}^*. \tag{3.35}$$

The next result discusses how a designer can tune the stopping criteria parameter δ in the 2HopDMAMPC Algorithm to approximate the final estimates to the optimal control input from the CMAMPC.

Lemma 4. *Consider the 2HopDMAMPC Algorithm and suppose Assumptions 2 holds.*

Then, given $\gamma > 0$, there exists $\delta > 0$ and m such that

$$\left\| \begin{bmatrix} \hat{\hat{u}}^i_{t:t+T-1}(m-1) \\ \hat{\hat{u}}^i_{t:t+T-1}(m) \end{bmatrix} - \begin{bmatrix} \hat{\hat{u}}^i_{t:t+T-1}(m-2) \\ \hat{\hat{u}}^i_{t:t+T-1}(m-1) \end{bmatrix} \right\| \leqslant \delta \|\varepsilon^i_t\|, \quad \forall i \;\Rightarrow\; \|\hat{u}_t(m) - u^*_t\| \leqslant \gamma \|\varepsilon_t\|. \quad \square \tag{3.36}$$

Proof. From Theorem 3 we have that there exists an iteration index m for which every agent satisfies a common stopping condition at line 13 of the 2HopDMAMPC Algorithm. Then, for the entire network,

$$\left\| \begin{bmatrix} \hat{\hat{u}}^1_{t:t+T-1}(m-1) \\ \hat{\hat{u}}^1_{t:t+T-1}(m) \\ \vdots \\ \hat{\hat{u}}^N_{t:t+T-1}(m-1) \\ \hat{\hat{u}}^N_{t:t+T-1}(m) \end{bmatrix} - \begin{bmatrix} \hat{\hat{u}}^1_{t:t+T-1}(m-2) \\ \hat{\hat{u}}^1_{t:t+T-1}(m-1) \\ \vdots \\ \hat{\hat{u}}^N_{t:t+T-1}(m-2) \\ \hat{\hat{u}}^N_{t:t+T-1}(m-1) \end{bmatrix} \right\| = \|w_{t:t+T-1}(m) - w_{t:t+T-1}(m-1)\| \leqslant \delta \|\varepsilon_t\|, \tag{3.37}$$

where $w_{t:t+T-1}$ is the state for the system in (3.28) that represents the evolution of the estimates. Denoting by $w^*_{t:t+T-1}$ its equilibrium we have that

$$\begin{aligned} w_{t:t+T-1}(m) - w^*_{t:t+T-1} &= \bar{\mathcal{M}} w_{t:t+T-1}(m-1) + \bar{\mathcal{P}}\varepsilon_t - (\bar{\mathcal{M}} w^*_{t:t+T-1} + \bar{\mathcal{P}}\varepsilon_t) \\ &= \bar{\mathcal{M}} w_{t:t+T-1}(m-1) - \bar{\mathcal{M}} w^*_{t:t+T-1}. \end{aligned} \tag{3.38}$$

Subtracting $w_{t:t+T-1}(m-1)$ and adding $w^*_{t:t+T-1}$ on both sides yields

$$\begin{aligned} w_{t:t+T-1}(m) - w_{t:t+T-1}(m-1) \\ &= \bar{\mathcal{M}} w_{t:t+T-1}(m-1) - w_{t:t+T-1}(m-1) - \bar{\mathcal{M}} w^*_{t:t+T-1} + w^*_{t:t+T-1} \\ &= (\bar{\mathcal{M}} - I)(w_{t:t+T-1}(m-1) - w^*_{t:t+T-1}). \end{aligned}$$

In Theorem 3 we showed that $\bar{\mathcal{M}}$ is Schur stable under Assumption 2, hence $\bar{\mathcal{M}} - I$ is invertible. Then,

$$w_{t:t+T-1}(m-1) - w^*_{t:t+T-1} = (\bar{\mathcal{M}} - I)^{-1}\big(w_{t:t+T-1}(m) - w_{t:t+T-1}(m-1)\big),$$

and using the relationship in (3.38) we obtain

$$w_{t:t+T-1}(m) - w^*_{t:t+T-1} = \bar{\mathcal{M}}(\bar{\mathcal{M}} - I)^{-1}\big(w_{t:t+T-1}(m) - w_{t:t+T-1}(m-1)\big).$$

Since the estimates of the control input are the output of the system in (3.28) then,

$$\hat{\bar{u}}_{t:t+T-1}(m) - u^*_{t:t+T-1} = \bar{\mathcal{C}}\bar{\mathcal{M}}(\bar{\mathcal{M}} - I)^{-1}\big(w_{t:t+T-1}(m) - w_{t:t+T-1}(m-1)\big).$$

Let $\mathcal{V} \in \mathbb{R}^{Nn_u \times TNn_u}$ the matrix that selects the first element of the control input sequence such that $\hat{\hat{u}}_t(m) = \mathcal{V}\hat{\hat{u}}_{t:t+T-1}(m)$. Then,

$$\hat{\hat{u}}_t(m) - u_t^* = \mathcal{V}\bar{\mathcal{C}}\bar{\mathcal{M}}\big(\bar{\mathcal{M}} - I)^{-1}(w_{t:t+T-1}(m) - w_{t:t+T-1}(m-1)\big).$$

Taking the norms and using (3.37) we have that for any $\gamma > 0$ we can pick

$$\delta = \gamma\|\mathcal{V}\bar{\mathcal{C}}\bar{\mathcal{M}}(\bar{\mathcal{M}} - I)^{-1}\|^{-1} \tag{3.39}$$

such that (3.36) holds. ■

When executing the 2HopDMAMPC Algorithm at every iteration m the stopping condition

$$\left\| \begin{bmatrix} \hat{\hat{u}}^i_{t:t+T-1}(m-1) \\ \hat{\hat{u}}^i_{t:t+T-1}(m) \end{bmatrix} - \begin{bmatrix} \hat{\hat{u}}^i_{t:t+T-1}(m-2) \\ \hat{\hat{u}}^i_{t:t+T-1}(m-1) \end{bmatrix} \right\| \leqslant \delta\|\varepsilon^i_t\|, \qquad \forall i \in \mathcal{N},$$

has to be checked. Alternatively, when the algorithm is initialized we can compute the number of iteration that the algorithm needs to execute to ensure that the resulting estimated control is sufficiently close to the centralized optimal one.

Corollary 2. *Consider the 2HopDMAMPC Algorithm and suppose that Assumptions 1 and 2 holds. Let γ be a user defined parameter. Then for all $m > m^*$ we have that*

$$\|\hat{\hat{u}}_t(m) - u_t^*\| \leqslant \gamma\|\varepsilon_t\|. \tag{3.40}$$

where

$$m^* = 1 + \frac{\log(\gamma) - \log(\|\mathcal{V}\bar{\mathcal{C}}\bar{\mathcal{M}}(\bar{\mathcal{M}} - I)^{-1}\|) + \log(\|\varepsilon_t\|) - \log(\|(\bar{\mathcal{M}} - I)w_{t:t+T-1}(0) + \bar{\mathcal{P}}\varepsilon_t\|)}{\log(\|\bar{\mathcal{M}}\|)}.$$

$\square$

Proof. The sufficient number of iterations m^* can be obtained by inspection of the solution of $w_{t:t+T-1}(m)$ given the initial condition $w_{t:t+T-1}(0)$. First note that the difference between two consecutive iterations is

$$\begin{aligned} w_{t:t+T-1}(m) - w_{t:t+T-1}(m-1) \\ &= \bar{\mathcal{M}}^m w_{t:t+T-1}(0) + \sum_{\tau=0}^{m-1} \bar{\mathcal{M}}^\tau \bar{\mathcal{P}}\varepsilon_t - \bar{\mathcal{M}}^{m-1} w_{t:t+T-1}(0) - \sum_{\tau=0}^{m-2} \bar{\mathcal{M}}^\tau \bar{\mathcal{P}}\varepsilon_t \\ &= \bar{\mathcal{M}}^{m-1}\big((\bar{\mathcal{M}} - I)w_{t:t+T-1}(0) + \bar{\mathcal{P}}\varepsilon_t\big). \end{aligned}$$

Suppose that for some $m > m^*$ we have that

$$\gamma\|\mathcal{V}\bar{\mathcal{C}}\bar{\mathcal{M}}(\bar{\mathcal{M}} - I)^{-1}\|^{-1}\|\varepsilon_t\| \geqslant \|\bar{\mathcal{M}}\|^{m-1}\|(\bar{\mathcal{M}} - I)w_{t:t+T-1}(0) + \bar{\mathcal{P}}\varepsilon_t\|, \tag{3.41}$$

where $\gamma > 0$ is a user defined parameter. Then,

$$\begin{aligned} \gamma\|\mathcal{V}\bar{\mathcal{C}}\bar{\mathcal{M}}(\bar{\mathcal{M}} - I)^{-1}\|^{-1}\|\varepsilon_t\| &\geqslant \|\bar{\mathcal{M}}\|^{m-1}\|(\bar{\mathcal{M}} - I)w_{t:t+T-1}(0) + \bar{\mathcal{P}}\varepsilon_t\| \\ &\geqslant \|\bar{\mathcal{M}}^{m-1}(\bar{\mathcal{M}} - I)w_{t:t+T-1}(0) + \bar{\mathcal{P}}\varepsilon_t\| \end{aligned}$$

$$= \|w_{t:t+T-1}(m) - w_{t:t+T-1}(m-1)\|.$$

From Lemma 4 we have that

$$\|w_{t:t+T-1}(m) - w_{t:t+T-1}(m-1)\| \leqslant \delta \|\varepsilon_t\|,$$

where $\delta \coloneqq \gamma \|\mathcal{V}\bar{\mathcal{C}}\bar{\mathcal{M}}(\bar{\mathcal{M}} - I)^{-1}\|^{-1}$ implies that (3.40) holds. The lower bound on the number of iterations is obtained by solving

$$\gamma \|\mathcal{V}\bar{\mathcal{C}}\bar{\mathcal{M}}(\bar{\mathcal{M}} - I)^{-1}\|^{-1} \|\varepsilon_t\| = \|\bar{\mathcal{M}}\|^{m^*-1} \|(\bar{\mathcal{M}} - I) w_{t:t+T-1}(0) + \bar{\mathcal{P}}\varepsilon_t\|$$

for m^*. ■

The previous results discussed how local stopping criteria or the number of iterations in the 2HopDMAMPC Algorithm can be chosen such that the estimates are arbitrarily close to the optimal control input sequence given by the CMAMPC problem. To study how the estimates obtained from the algorithm affect the stability of the system, we express the closed-loop system in (3.30) as

$$\begin{aligned}\varepsilon_{t+1} &= \bar{A}\varepsilon_t + \bar{B}u_t^* + \bar{B}\hat{\hat{u}}_t(m) - Bu_t^* \\ &= \bar{A}^*\varepsilon_t + \bar{B}(\bar{\mathcal{K}}(m) - \mathcal{K}^*)\varepsilon_t + \bar{B}\mathcal{V}\bar{\mathcal{C}}\bar{\mathcal{M}}^m w_{t:t+T-1}(0)\end{aligned}$$

where we use definitions for the controls u^* and $\hat{\hat{u}}_t(m)$ from (3.19) and (3.31), respectively, and where $\bar{A}^* \coloneqq \bar{A} + \bar{B}\mathcal{K}^*$. For simplicity, we assume that for every agent and at every time step the 2HopDMAMPC Algorithm is initialized at zero (lines 3 and 6), that is,

$$w_{t:t:T-1}(0) = 0 \qquad \forall t \geqslant 0,$$

so the closed-loop system (3.30) can be simplified to

$$\varepsilon_{t+1} = \big(\bar{A}^* + \bar{B}(\bar{\mathcal{K}}(m) - \mathcal{K}^*)\big)\varepsilon_t. \tag{3.42}$$

We are ready to present the main result of this section.

Theorem 4 (Closed-loop stability using the 2HopDMAMPC Algorithm)**.** *Consider the closed-loop system in* (3.30) *and suppose that Assumptions 1 and 2 hold. Let the terminal costs for the distributed optimization in* (3.22) *be chosen according to the statement in Theorem 2. Then for any stopping criteria parameter δ in the 2HopDMAMPC Algorithm such that*

$$\delta \leqslant \frac{1}{2}\|\bar{A}^{*\top}\Phi\bar{B}\|^{-1}\|\mathcal{V}\bar{\mathcal{C}}\bar{\mathcal{M}}(\bar{\mathcal{M}} - I)^{-1}\|^{-1}, \tag{3.43}$$

where $\Phi > 0$ is such that for some $\Lambda > 0$

$$A^{*\top}\Phi\bar{A}^* - \Phi + \Lambda = 0, \tag{3.44}$$

then the origin is asymptotically stable. □

Proof. The asymptotic stability of (3.42) (or (3.30)) can be established by showing that for some $\Phi > 0$ the following inequality holds

$$\big(\bar{A}^* + \bar{B}(\bar{\mathcal{K}}(m) - \mathcal{K}^*)\big)^\top \Phi \big(\bar{A}^* + \bar{B}(\bar{\mathcal{K}}(m) - \mathcal{K}^*)\big) - \Phi < 0. \tag{3.45}$$

From Theorem 2 we have that the system in (3.21) is asymptotically stable, hence there exists a unique $\Phi > 0$ that is the solution of (3.44) for any $\Lambda > 0$. Expanding the left-hand side of (3.45) and using (3.44) we have that a sufficient condition for stability is

$$\bar{A^*}^\top \Phi \bar{B}(\mathcal{K}(m) - \mathcal{K}^*) + (\mathcal{K}(m) - \mathcal{K}^*)^\top \bar{B}^\top \Phi \bar{A}^* + (\mathcal{K}(m) - \mathcal{K}^*)^\top \bar{B}^\top \Phi \bar{B}(\mathcal{K}(m) - \mathcal{K}^*) < \Lambda,$$

and since the last term on the left-hand side of the above inequality is positive semi-definite the inequality can be simplified to

$$\bar{A^*}^\top \Phi \bar{B}(\mathcal{K}(m) - \mathcal{K}^*) + (\mathcal{K}(m) - \mathcal{K}^*)^\top \bar{B}^\top \Phi \bar{A}^* < \Lambda.$$

This inequality can be enforce by satisfying the following relationship

$$\frac{1}{2}\|\bar{A}^{*\top} \Phi \bar{B}\|^{-1} \underline{\lambda}(\Lambda) > \|\mathcal{K}(m) - \mathcal{K}^*\|.$$

Multiplying by $\|\varepsilon_t\|$ on both sides yields

$$\begin{aligned}\frac{1}{2}\|\bar{A}^{*\top}\Phi\bar{B}\|^{-1}\underline{\lambda}(\Lambda)\|\varepsilon_t\| &> \|\bar{\mathcal{K}}(m)-\mathcal{K}^*\|\|\varepsilon_t\| \\ &\geqslant \|(\bar{\mathcal{K}}(m)-\mathcal{K}^*)\varepsilon_t\| \\ &\geqslant \|\hat{\bar{u}}_t(m)-u_t^* - \mathcal{V}\bar{\mathcal{C}}\bar{\mathcal{M}}w_{t:t+T-1}(0)\| \\ &= \|\hat{\bar{u}}_t(m)-u_t^*\|,\end{aligned}$$

where we use the fact that 2HopDMAMPC Algorithm is initialized with null estimates. Define

$$\gamma \coloneqq \frac{1}{2}\|\bar{A}^{*\top}\Phi\bar{B}\|^{-1}\underline{\lambda}(\Lambda),$$

such that

$$\|\hat{\bar{u}}_t(m)-u_t^*\| \leqslant \gamma\|\varepsilon_t\|. \tag{3.46}$$

Then, from Lemma 4 we have that for some δ satisfying (3.43)

$$\left\|\begin{bmatrix}\hat{\bar{u}}^i_{t:t+T-1}(m-1)\\ \hat{\bar{u}}^i_{t:t+T-1}(m)\end{bmatrix}-\begin{bmatrix}\hat{\bar{u}}^i_{t:t+T-1}(m-2)\\ \hat{\bar{u}}^i_{t:t+T-1}(m-1)\end{bmatrix}\right\| \leqslant \delta\|\varepsilon^i_t\|, \quad \forall i \quad \Rightarrow \ \|\hat{\bar{u}}_t(m)-u_t^*\| \leqslant \gamma\|\varepsilon_t\|,$$

and we conclude that the origin of (3.30) is globally asymptotically stable. ■

Remark 7. Instead of Lemma 4, the stability of the closed-loop system in (3.30) can be established by use of Corollary 2. There, we showed that there exists a sufficient number of iterations that the 2HopDMAMP Algorithm needs to execute to ensure that the resulting estimate is γ-close to the centralized optimal control, i.e., that (3.46) holds.

□

3.3.4 The 1-Hop Partial Optimization

In this section, we consider local optimizations that could be solved with 1-hop neighbor information, rather than the 2-hop information required before. For each $i \in \mathcal{N}$, we now consider the following local optimization

$$\min_{u^i_{t:t+T-1}} \tilde{J}^i(u^i_{t:t+T-1}; \varepsilon^i_t, u^{-i_1}_{t:t+T-1}), \tag{3.47}$$

subject to the dynamics

$$\varepsilon^i_{k+1} = A\varepsilon^i_k - (g^i + d^i)Bu^i_k + B\sum_{j\in\mathcal{N}_i} u^j_k, \qquad \forall k \in \{t, t+1, \ldots, t+T-1\},$$

for agent i. The local cost functions are defined by

$$\begin{aligned} J^i(u^i_{t:t+T-1}; \varepsilon^i_t) &\coloneqq \frac{1}{2}\sum_{k=t}^{t+T-1}({\varepsilon^i_k}^\top Q\varepsilon^i_k + {u^i_k}^\top Ru^i_k) + \frac{1}{2}{\varepsilon^i_{t+T}}^\top Q_f\varepsilon^i_{t+T} \\ &= \frac{1}{2}({\varepsilon^i_{t+1:t+T}}^\top \mathcal{Q}\varepsilon^i_{t+1:t+T} + {u^i_{t:t+T-1}}^\top \mathcal{R}u^i_{t:t+T-1} + {\varepsilon^i_t}^\top Q\varepsilon^i_t). \end{aligned}$$

Note that $u^i_{t:t+T-1}$ is still the optimization variable for the local problem defined in

(3.47), but the solution to this problem now depends only on the local tracking error ε_t^i and on the variables $u_{t:t+T-1}^{-i_1}$ that are associated with the 1-hop neighbors of i.

Denoting by $\tilde{u}_{t:t+T-1}^{i*}$ the solutions to (3.47), the first-order optimality condition for this problem is

$$\nabla_{u_{t:t+T-1}^i} \tilde{J}^i(u_{t:t+T-1}^i; \varepsilon_t^i, u_{t:t+T-1}^{-i_1})\Big|_{u_{t:t+T-1}^i = \tilde{u}_{t:t+T-1}^{i*}} = 0$$

which implies

$$\big((g^i + d^i)^2 \mathcal{B}^\top \mathcal{Q}\mathcal{B} + \mathcal{R}\big)\tilde{u}_{t:t+T-1}^{i*} - (g^i + d^i)\mathcal{B}^\top \mathcal{Q}\mathcal{B} \sum_{j \in \mathcal{N}_i} u_{t:t+T-1}^j - (g^i + d^i)(\mathcal{B}^\top \mathcal{Q}\mathcal{A})\varepsilon_t^i = 0, \tag{3.48}$$

and for all the conditions (3.48), one for each agent $i \in \mathcal{N}$, we have that

$$\begin{aligned} &\nabla_{u_{t:t+T-1}} \tilde{J}(u_{t:t+T-1}; \varepsilon_t)\Big|_{u_{t:t+T-1} = \tilde{u}_{t:t+T-1}^*} \\ &\qquad := \left[\nabla_{u_{t:t+T-1}^i} \tilde{J}^i(u_{t:t+T-1}^i; \varepsilon_t^i, u_{t:t+T-1}^{-i_1})\right]\Big|_{u_{t:t+T-1} = \tilde{u}_{t:t+T-1}^*} = 0. \end{aligned}$$

This first-order optimality condition can be expressed in vector form as

$$\big(((G + D)\bar{L}) \otimes (\mathcal{B}^\top \mathcal{Q}\mathcal{B}) + I_N \otimes \mathcal{R}\big)\tilde{u}_{t:t+T-1}^* - \big((G + D) \otimes (\mathcal{B}^\top \mathcal{Q}\mathcal{A})\big)\varepsilon_t = 0. \tag{3.49}$$

Unlike the 2-hop algorithm, these first-order optimality conditions for the local optimization (3.47) now differ from the centralized optimality conditions in (3.18). However, the following result shows that (3.49) still has a unique solution.

Proposition 2. *Suppose that Assumption 2 holds and let* $\mathcal{R} = rI_{Tn_u}$, $r > 0$. *Then the optimal control sequence obtained from solving* (3.47) *is unique and equal to*

$$\tilde{u}^*_{t:t+T-1} = (((G+D)\bar{L}) \otimes (\mathcal{B}^\top \mathcal{Q}\mathcal{B}) + I_N \otimes \mathcal{R})^{-1}((G+D) \otimes (\mathcal{B}^\top \mathcal{Q}\mathcal{A}))\varepsilon_t. \quad \square \tag{3.50}$$

Proof. The proof is similar to that found in Lemma 3 and it suffices to show that the eigenvalues of $(G + D)\bar{L}$ are positive. Let v and λ be an arbitrary eigenvector and eigenvalue pair of $(G + D)\bar{L}$. Then,

$$(G + D)\bar{L}v = \lambda v \Leftrightarrow \bar{L}v = \lambda(G + D)^{-1}v \Rightarrow \frac{v^\top \bar{L}v}{v^\top (G + D)^{-1}v} = \lambda > 0,$$

where the inequality comes from the fact that $\bar{L} > 0$ and $G + D > 0$. Finally, solving (3.49) for $\tilde{u}^*_{t:t+T-1}$ we obtain (3.50). ■

Remark 8. For the "cheap control" case, i.e., $\mathcal{R} = 0$, and under the assumption that the matrix $\mathcal{B}$ is full column rank, one can show that (3.50) minimizes (3.17). This also happens in the case when $r \to \infty$, $\mathcal{R} = rI_{n_u}$. In general, between these extremes (3.50) yields a suboptimal result, but as we see in the simulations, the optimality gap is often small. □

3.3.5 The 1HopDMAMPC Algorithm

The local costs in (3.47) are the basis for the 1-Hop Distributed Multi-Agent Model Predictive Control (1HopDMAMPC) Algorithm that requires only 1-hop information. The variable $\hat{\tilde{u}}^i_{t:t+T-1}(m)$ should be viewed as an estimate of $\tilde{u}^{i*}_{t:t+T-1}$ computed at iteration m.

Algorithm 3 1HopDMAMPC Algorithm for agent i

Require: a tolerance $\delta > 0$ and local state ε^i_t

1: Broadcast ε^i_t to neighbors

2: $m \leftarrow 0$

3: Initialize $\hat{\tilde{u}}^i_{t:t+T-1}(m)$

4: **repeat**

5: Broadcast $\hat{\tilde{u}}^i_{t:t+T-1}(m)$ to neighbors

6: $\hat{\tilde{u}}^i_{t:t+T-1}(m+1) \leftarrow \arg\min_{u^i_{t:t+T-1}} \tilde{J}^i\big(u^i_{t:t+T-1}; \varepsilon^i_t, \hat{\tilde{u}}^{-i_1}_{t:t+T-1}(m)\big)$

7: $m \leftarrow m+1$

8: $error \leftarrow \|\hat{\tilde{u}}^i_{t:t+T-1}(m) - \hat{\tilde{u}}^i_{t:t+T-1}(m-1)\|$

9: **until** $error \leqslant \delta \|\varepsilon^i_t\|$

return $\hat{\tilde{u}}^i_{t:t+T-1}(m)$

Similarly to the previous section, we revisit the first-order optimality condition to investigate the convergence of the estimates computed using the 1HopDMAMPC Algorithm.

At each iteration of the algorithm we have that the following equation must be satisfied

$$\begin{aligned}\big((g^i + d^i)^2\mathcal{B}^\top\mathcal{Q}\mathcal{B} + \mathcal{R}\big)\hat{\hat{u}}^i_{t:t+T-1}(m+1) - (g^i + d^i)\mathcal{B}^\top\mathcal{Q}\mathcal{B}\sum_{j\in\mathcal{N}_i}\hat{\hat{u}}^j_{t:t+T-1}(m)\\ - (g^i + d^i)(\mathcal{B}^\top\mathcal{Q}\mathcal{A})\varepsilon^i_t = 0\end{aligned}$$

or, in vector form for every agent,

$$\begin{aligned}\big((G+D)^2\otimes(\mathcal{B}^\top\mathcal{Q}\mathcal{B}) + I_N\otimes\mathcal{R}\big)\hat{\hat{u}}_{t:t+T-1}(m+1) - \big(((G+D)A_d)\otimes(\mathcal{B}^\top\mathcal{Q}\mathcal{B})\big)\hat{\hat{u}}_{t:t+T-1}(m)\\ - ((G+D)\otimes(\mathcal{B}^\top\mathcal{Q}\mathcal{A}))\varepsilon_t = 0. \quad (3.51)\end{aligned}$$

The next theorem establishes the convergence of the iterations given by (3.51).

Theorem 5 (Convergenge of the 1HopDMAMPC Algorithm)**.** *Consider the 1HopDMA-MPC Algorithm with dynamics given by* (3.51) *and let Assumption 2 hold. Then the local estimates* $\hat{\hat{u}}_{t:t+T-1}$ *converge to* (3.50). □

The following lemma addresses the stability of first-order systems of a difference equation similar to (3.51).

Lemma 5. *Consider the following discrete-time system*

$$Ax(m+1) - Bx(m) - c = 0$$

where $x, c \in \mathbb{R}^n$ $A, B \in \mathbb{R}^{n\times n}$. Assume that $A > 0$ and $2A \pm (B + B^\top) > 0$. Then the system is asymptotically stable.

Proof of Lemma 5. Let v and λ be an arbitrary eigenvector and eigenvalue pair of $A^{-1}B$. Then,

$$A^{-1}Bv = \lambda v \Leftrightarrow Bv = \lambda Av \Rightarrow \frac{v^\top Bv}{v^\top Av} = \lambda \Leftrightarrow \frac{v^\top (B + B^\top)v}{v^\top 2Av} = \lambda.$$

From the assumption on the lemma, we have that $|\lambda| < 1$, and the state asymptotically converges to $x^* := (A - B)^{-1}c$. ∎

Proof of Theorem 5. The proof is a direct application of Lemma 5 to the system in (3.51). We first note that $(G + D)^2 \otimes (\mathcal{B}^\top \mathcal{Q}\mathcal{B}) + I_N \otimes \mathcal{R} > 0$ since $G + D > 0$, $\mathcal{B}^\top \mathcal{Q}\mathcal{B} \geqslant 0$, and $\mathcal{R} > 0$. To check if the second matrix inequality holds we write

$$\begin{aligned} 2\big((G + D)^2 \otimes (\mathcal{B}^\top \mathcal{Q}\mathcal{B}) + I_N \otimes \mathcal{R}\big) - \big((A_d(G + D) + (G + D)A_d) \otimes (\mathcal{B}^\top \mathcal{Q}\mathcal{B})\big) = \\ (\bar{L}(G + D) + (G + D)\bar{L}) \otimes (\mathcal{B}^\top \mathcal{Q}\mathcal{B}) + 2I_N \otimes \mathcal{R}. \end{aligned} \tag{3.52}$$

From the discussion in Proposition 2 we concluded that the eigenvalues of $(G + D)\bar{L}$ are positive and $B^\top \mathcal{Q}\mathcal{B} \geqslant 0$, hence (3.52) is positive definite. To show that

$$\begin{aligned} 2((G + D)^2 \otimes (\mathcal{B}^\top \mathcal{Q}\mathcal{B}) + I_N \otimes \mathcal{R}) + ((A_d(G + D) + (G + D)A_d) \otimes (\mathcal{B}^\top \mathcal{Q}\mathcal{B})) = \\ ((Ad + G + D)(G + D) + (G + D)(Ad + G + D)) \otimes (\mathcal{B}^\top \mathcal{Q}\mathcal{B}) + I_N \otimes \mathcal{R} \end{aligned}$$

is positive definite, we use the fact that the eigenvalues of $(A_d + G + D)(G + D)$ are positive and that $\mathcal{B}^\top \mathcal{Q} \mathcal{B} \geqslant 0$. We conclude that the system in (3.51) is asymptotically stable and converges to the unique equilibrium (3.50). ■

3.4 Numerical Example

We now illustrate our algorithms to solve optimizations associated with a model predictive control problem of the form (3.5). At each time-step t, we execute the 2HopDMAMPC and the 1HopDMAMPC Algorithms to estimate the centralized optimal control input sequence obtained from the CMAMPC and from this sequence we utilize only the first value of control input to drive the followers and repeat this process at the next time step with a new value of the state.

We generated random graphs with arbitrary topology and number of nodes satisfying Assumption 2. For our simulations, we use one such graphs with 10 follower agents, where only one of them is connected to the single leader agent. The maximum node degree in this graph is 3. We consider the synchronization of an unstable third-order linear systems like (3.1) with

$$A = \begin{bmatrix} 2 & 0 & 1 \\ 0 & 1 & 1 \\ 1 & -2 & 1 \end{bmatrix}, \ B = \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix}.$$

The initial state for each agent is randomly chosen and we select the weights on the cost as $Q = 10I_3$, $Q_f = 100I_3$ $R = 10$, and $T = 15$. The optimizations were solved using `TensCalc` [24], a toolbox that generates efficient solvers for large-scale optimization problems.

Figure 3.1 shows a typical evolution of the norm of the tracking error ε_t for one of the agents, comparing both algorithms, and showing that synchronization is achieved.

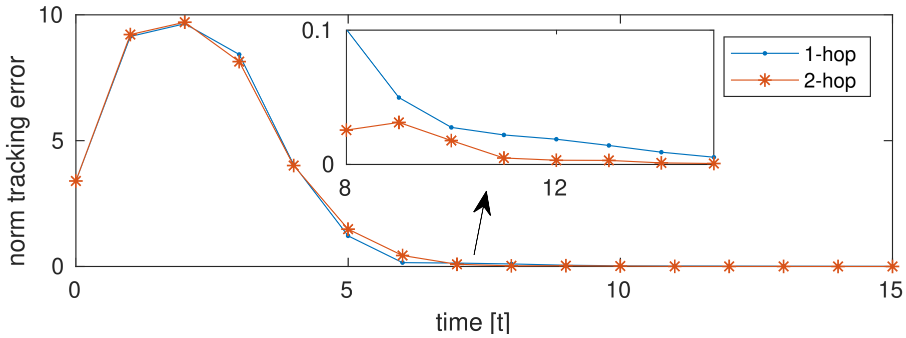


Figure 3.1: Comparison between the tracking error for a given follower agent for each of the two algorithms.

To illustrate Remark 8, we simulate various values of the weight R in the control input for a particular time instant of the simulation, comparing the cost from converged estimates obtained from the 1HopDMAMPC Algorithm versus the optimal cost in (3.14). Figure 3.2 confirms that the gap is small, and that in the extreme cases (i.e., $R = 0$ and $R = \rho$, with $\rho \to \infty$) the estimates obtained from the 1-hop algorithm are indeed optimal solutions to the optimization (3.5).

3.5 Conclusion

In this chapter we considered a leader-follower synchronization problem where the followers' objective is to track the leader state. We presented two distributed algorithms to

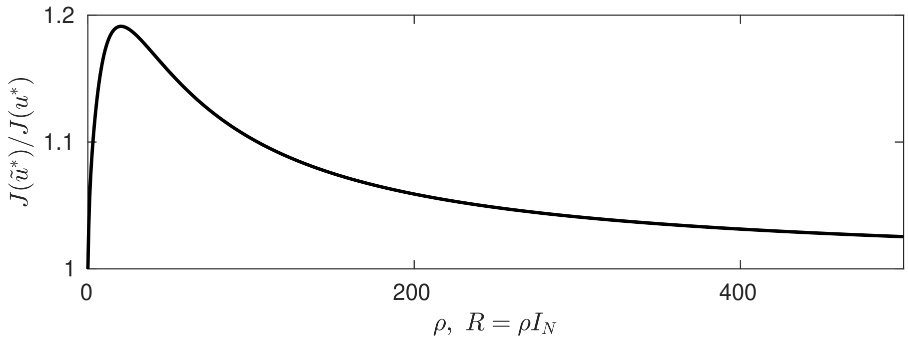


Figure 3.2: Ratio between the cost from the converged estimated of the control input using 1-hop neighbor information, and the optimal cost for various values of the weighting matrix R.

estimate the optimal control sequence obtained from a centralized finite horizon quadratic optimization of the type found in model predictive control problems for multi-agent coordination.

When 2-hop information is considered we showed that the estimates obtained from the distributed algorithm converge to the centralized solution. We then introduced a state-dependent stopping criteria and a minimum number of iterations for the algorithm that guarantee that when the estimated control is used the leader-follower synchronization is achieved. When only 1-hop neighbor information is considered, a second algorithm was proposed that can still be shown to converge, but yields suboptimal results. Simulations verified the closed-loop stability numerically for a multi-agent system with unstable dynamics.

Chapter 4

Conclusion

4.1 Contribution of this Dissertation

In this dissertation we discussed a new approach for solving optimization problems related to estimation and control of multi-agent systems. The main contributions are three distributed and iterative algorithms that compute the solution of centralized optimization problems. These algorithms utilize only local information to iteratively compute a sequence of smaller optimization problems. Theoretical results related to the convergence of the algorithms and the closed-loop stability of the controlled systems are presented in the context of two distinct applications. The chapters' individual conclusions are summarize next.

Distributed Localization in Sensor Networks

In Chapter 2 we addressed the problem of parameter estimation in sensor networks. We showed how several types of inter-sensor measurements can be used to formulate maximum likelihood estimation problems that are based in optimization.

We proposed a distributed algorithm that utilizes locally available measurements to iteratively estimate the solution of the maximum likelihood estimation problem. The method relies on the communication between the nodes in the network to solve a sequence of smaller optimization problems locally. To study the convergence of the estimates, we linearized the dynamics that describe the evolution of the algorithm and derived sufficient condition for local asymptotic stability. The method was validated in simulation, for a given localization problem, and in hardware for a combined localization and clock synchronization problem. The simulations showed that in general the convergence to the optimal solution occurs regardless of how the algorithm is initialized. The efficacy of our algorithms was also confirmed experimentally, and the results showed that our method outperformed other distributed approaches tested.

Distributed Coordination for Multi-Agent Systems

Chapter 3 discussed a coordination problem for multi-agent systems where the objective of a collection of follower agents is to track a single leader agent. We introduced an MPC to this problem, and derived sufficient conditions for synchronization to occur.

We then presented two algorithms to estimate the solution to the MPC problem. The methods we developed are distributed, requiring only local information to iteratively solve a sequence of optimizations, similar to what was presented in Chapter 2.

The first method relies on 2-hop neighbor information, whereas the second needs only 1-hop information. For the 2-hop method we showed that the estimates converge asymptotically to the optimal solution for the original MPC problem and we derived practical conditions to terminate the algorithm in finite number of iterations, such that the generated control inputs still yield synchronization. When only 1-hop information is considered, the algorithm converged to a subopotimal result, but in some cases the optimality gap is small and even zero. Simulations for a multi-agent system with unstable dynamics illustrated the theoretical results obtained in this chapter.

4.2 Future work

There are several extensions that can be developed to improve upon the work in this dissertation. Estimation in the presence of adversarial attacks, packet losses, and asynchronous communication between the nodes are common scenarios in wireless networked sensors and if formally addressed would significantly improve the security and reliability of the developed algorithm. From an analysis point of view, simulations and preliminary results suggest a relationship between the graph topology and the region and rate of convergence of the localization algorithm. It would be interesting to investigate sufficient conditions on the network topology for global convergence of the estimates.

For the type of problems presented in Chapter 3, we are currently studying distributed algorithms for coordination of multi-agent over switching graphs, such as those arising in ad hoc or unreliable networks. In this scenario the challenge comes from the unpredictable nature of the switches that contrasts with the model-based prediction nature of the MPC. The proposed method could also be greatly improved if constraints on the state and control input are considered in the optimization problem, as well as nonlinear models and more general cost functions.

Finally, several other problems related to estimation and control could benefit from the use of the distributed methods presented in this dissertation. Examples include the optimal power flow problem and the synchronization of oscillators in problems related to power systems, vehicle platooning, as well as general formation control for autonomous vehicles, and chemical process control. The possibilities of new results and applications in this area are timely and exciting.

Appendix A

Proofs of the technical lemmas in Remark 1

Lemma 6. *Given a set of column vectors $s_1, s_2, \ldots, s_m \in \mathbb{R}^n$, the $n \times n$ matrix*

$$S = \sum_{i=1}^{m} s_i s_i^\top = \begin{bmatrix} s_1 & s_2 & \cdots \end{bmatrix}_{n \times m} \begin{bmatrix} s_1^\top \\ s_2^\top \\ \vdots \\ s_m^\top \end{bmatrix}_{m \times n}$$

is nonsingular if and only if the m vectors span $\mathbb{R}^n$ (i.e., n of the vectors are linearly independent). Moreover, if $s_i^\top = \begin{bmatrix} w_i^\top & 1 \end{bmatrix}$, $w_i \in \mathbb{R}^{n-1}$, then S is nonsingular if and only if there exists a set $\mathcal{L}$ of $n-1$ linearly independent vectors w_i and one vector

$w_k \neq w_i \ \forall w_i \in \mathcal{L}$. $\square$

Lemma 7. *Given three points $p_0, p_1, p_2 \in \mathbb{R}^2$ in the plane, the two vectors*

$$s_1 \coloneqq p_1 - p_0, \qquad s_2 \coloneqq p_2 - p_0$$

are linearly independent if and only if the three points are not colinear. $\square$

Proof of Lemma 7. Linearly independence is equivalent to one of the vectors (say s_2 for concreteness) being alinged with the other, i.e.,

$$\exists \alpha_1 \in \mathbb{R} : p_2 - p_0 = \alpha_1(p_1 - p_0) \quad \Leftrightarrow \quad \exists \alpha_1 \in \mathbb{R} : p_2 = (1 - \alpha_1)p_0 + \alpha_1 p_1$$

which is precisely equivalent to one of the points being in the line defined by the other two points. $\blacksquare$

Lemma 8. *Given four points $p_0, p_1, p_2, p_3 \in \mathbb{R}^3$ in the plane, the three vectors*

$$s_1 \coloneqq p_1 - p_0, \qquad s_2 \coloneqq p_2 - p_0, \qquad s_3 \coloneqq p_3 - p_0$$

are linearly independent if and only if the four points are not coplanar. $\square$

Proof of Lemma 8. Linearly independence is equivalent to one of the vectors (say s_3 for concreteness) being a linear combination of the remaining two, i.e.,

$$\exists \alpha_1, \alpha_2 \in \mathbb{R} : p_3 - p_0 = \alpha_1(p_1 - p_0) + \alpha_2(p_2 - p_0) \quad \Leftrightarrow$$
$$\Leftrightarrow \quad \exists \alpha_1, \alpha_2 \in \mathbb{R} : p_3 = (1 - \alpha_1 - \alpha_2)p_0 + \alpha_1 p_1 + \alpha_2 p_2$$

which is precisely equivalent to one of the points being in the plane defined by the other two points. ■

Lemma 9. *Given five points $p_0, p_1, p_2, p_3, p_4 \in \mathbb{R}^3$ in the plane, the four vectors*

$$s_1 := \begin{bmatrix} p_1 - p_0 \\ 1 \end{bmatrix}, \quad s_2 := \begin{bmatrix} p_2 - p_0 \\ 1 \end{bmatrix}, \quad s_3 := \begin{bmatrix} p_3 - p_0 \\ 1 \end{bmatrix}, \quad s_4 := \begin{bmatrix} p_4 - p_0 \\ 1 \end{bmatrix}$$

are linearly independent if and only if the points $p_1, p_2, p_3, p_4 \in \mathbb{R}^3$ are not coplanar. □

Proof of Lemma 9. Linearly independence is equivalent to one of the vector (say s_4 for concreteness) being a linear combination of the remaining three, i.e.,

$$\exists \alpha_1, \alpha_2, \alpha_3 \in \mathbb{R} : \begin{bmatrix} p_4 - p_0 \\ 1 \end{bmatrix} = \begin{bmatrix} \alpha_1(p_1 - p_0) \\ \alpha_1 \end{bmatrix} + \begin{bmatrix} \alpha_2(p_2 - p_0) \\ \alpha_2 \end{bmatrix} + \begin{bmatrix} \alpha_3(p_1 - p_0) \\ \alpha_3 \end{bmatrix} \quad \Leftrightarrow$$
$$\Leftrightarrow \quad \exists \alpha_1, \alpha_2, \alpha_3 \in \mathbb{R} : p_4 = \alpha_1 p_1 + \alpha_2 p_2 + \alpha_3 p_3, \ \alpha_1 + \alpha_2 + \alpha_3 = 1$$

which is precisely equivalent to one of the points in p_1, p_2, p_3, p_4 being in the plane defined by the remaining three points. ■

Bibliography

[1] N. M. M. de Abreu, “Old and new results on algebraic connectivity of graphs,” *Linear Algebra and its Applications*, vol. 423, issue 1, pp. 53-73, 2007.

[2] A. Alanwar, H. Ferraz, K. Hsieh, R. Thazhath, P. Martin, J. Hespanha, M. Srivastava. “D-SLATS: Distributed Simultaneous Localization and Time Synchronization,” in *Proc. of the ACM MOBIHOC*, June 2017.

[3] B. D. Anderson, B. D., Belhumeur, P. N., Eren, T., Goldenberg, D. K., Morse, A. S., Whiteley, W., and Yang, Y. R. “Graphical properties of easily localizable sensor networks.” *Wireless Networks* no. 15, vol.2, pp 177-191, 2009.

[4] J. Aspnes et al., “A Theory of Network Localization,” in *IEEE Transactions on Mobile Computing*, vol. 5, no. 12, pp. 1663-1678, Dec. 2006.

[5] P. Barooah and J. P. Hespanha, “Estimation on graphs from relative measurements,” in *IEEE Control Systems*, vol. 27, no. 4, pp. 57-74, Aug. 2007.

[6] Bitcraze. *CrazyFlie 2.0*, 2017 (accessed: 2017-01-01) `https://www.bitcraze.io/`.

[7] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, “Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers”, *Found. Trends Mach. Learn.*, vol. 3, pp. 1-122, 2011.

[8] Y. Cao, W. Yu, W. Ren and G. Chen, “An Overview of Recent Progress in the Study of Distributed Multi-Agent Coordination,” in *IEEE Transactions on Industrial Informatics*, vol. 9, no. 1, pp. 427-438, Feb. 2013.

[9] F. S. Cattivelli and A. H. Sayed, “Distributed nonlinear Kalman filtering with applications to wireless localization,” *2010 IEEE International Conference on Acoustics, Speech and Signal Processing*, Dallas, TX, pp. 3522-3525, 2010.

[10] C.-Y. Chong, S. Kumar, “Sensor networks: evolution,opportunities, and challenges”, in *Proceedings of the IEEE,* vol. 91, no. 8, 12471256, Aug. 2003.

[11] P. D. Christofides, R. Scattolini, D. M. de la Pena, and J. Liu, “Distributed model predictive control: A tutorial review and future research directions”, *Computers & Chemical Engineering*, vol. 51, pp. 21-41, 2013.

[12] D. A. Copp, K. G. Vamvoudakis, and J. P. Hespanha, "Distributed Output-Feedback Model Predictive Control, *Systems and Control Letters*, vol. 127, pp. 52-59, 2019.

[13] R. T. Collins, A. J. Lipton, H. Fujiyoshi, and T. Kanade, "Algorithms for cooperative multisensor surveillance," in *Proceedings of the IEEE*, vol. 89, no. 10, pp. 1456-1477, Oct. 2001.

[14] Decawave. *DW1000 IR-UWB description*, 2017 (accessed: 2017-01-01). `http://www.decawave.com/products/dw1000`.

[15] L. Doherty, K. S. J. Pister and L. El Ghaoui, "Convex position estimation in wireless sensor networks," *Proceedings IEEE INFOCOM 2001. Conference on Computer Communications*. vol.3, pp. 1655-1663, 2001.

[16] W. B. Dunbar and R. M. Murray, "Distributed receding horizon control for multi-vehicle formation stabilization", *Automatica*, vol. 42, no. 4, pp. 549-558, 2006.

[17] H. Ferraz, A. Alanwar, M. Srivastava and J. P. Hespanha, "Node localization based on distributed constrained optimization using Jacobi's method," in *IEEE 56th Annual Conference on Decision and Control (CDC)*, Melbourne, VIC, pp. 3380-3385, 2017.

[18] H. Ferraz, J. Hespanha. "Iterative algorithms for distributed leader-follower model predictive control". in *IEEE 58th Annual Conference on Decision and Control (CDC)*, 2019. To appear

[19] H. Ferraz, G. Yang, J. P. Hespanha, "Distributed algorithms for model predictive control based coordination of multi-agent systems", in preparation.

[20] E. Franco, L. Magni, T. Parisini, M. M. Polycarpou, and D. M. Raimondo, "Cooperative Constrained Control of Distributed Agents With Nonlinear Dynamics and Delayed Information Exchange: A Stabilizing Receding-Horizon Approach," *IEEE Transactions on Automatic Control*, vol. 53, no. 1, pp. 324-338, Feb. 2008.

[21] B. Gharesifard and J. Cortés, "Distributed Continuous-Time Convex Optimization on Weight-Balanced Digraphs," *IEEE Transactions on Automatic Control*, vol. 59, no. 3, pp. 781-786, March 2014.

[22] G. Grimm, M. J. Messina, S. E. Tuna, and A. R. Teel, "Model predictive control: for want of a local control Lyapunov function, all is not lost," *IEEE Transactions on Automatic Control*, vol. 50, no. 5, pp. 546-558, May 2005.

[23] J. K. Hale, *Ordinary Differential Equations*, Krieger Publishing Company, 1980.

[24] J. P. Hespanha, "`TensCalc` - A toolbox to generate fast code to solve nonlinear constrained minimizations and compute Nash equilibria," *Technical report*, `https://github.com/hespanha/tenscalc`, 2017.

[25] R. A. Horn and C. R. Johnson, *Matrix Analysis*. Cambridge University Press, 1985

[26] X. Ji and H. Zha, “Sensor positioning in wireless ad-hoc sensor networks using multidimensional scaling,” *IEEE INFOCOM 2004*, vol.4, pp. 2652-2661, 2004,

[27] B. Johansson, M. Rabi, and M. Johansson, “A randomized incremental subgradient method for distributed optimization in networked systems, *SIAM Journal on Control and Optimization*, vol. 20, no. 3, pp. 11571170, 2009.

[28] U. A. Khan and J. M. F. Moura, “Distributing the Kalman Filter for Large-Scale Systems,” in *IEEE Transactions on Signal Processing*, vol. 56, no. 10, pp. 4919-4935, Oct. 2008.

[29] S. S. Kia, J. Cortés, S. Martínez, “Distributed convex optimization via continuous-time coordination algorithms with discrete-time communication”, *Automatica*, vol. 55, pp. 254-264, May 2015.

[30] H. Li and Y. Shi, “Robust Distributed Model Predictive Control of Constrained Continuous-Time Nonlinear Systems: A Robustness Constraint Approach,” *IEEE Transactions on Automatic Control*, vol. 59, no. 6, pp. 1673-1678, June 2014.

[31] G. Mao, B. Fidan, B.D.O. Anderson, “Wireless sensor network localization techniques”, *Computer Networks*, vol. 51, issue 10, 11, pp. 2529-2553, July 2007.

[32] M. A. Muller, M. Reble, and F. Allgower, “Cooperative control of dynamically decoupled systems via distributed model predictive control,” *International Journal of Robust and Nonlinear Control*, vol. 22, Issue 12, pp. 1376-1397, 2012.

[33] R. Nagpal,H. Shrobe, and J. Bachrach “Organizing a global coordinate system from local information on an ad hoc sensor network”. *In Information Processing in Sensor Networks*, pp. 333-348, Springer Berlin Heidelberg, 2003.

[34] A. Nedíc and A. Ozdaglar, “Distributed subgradient methods for multiagent optimization, *IEEE Transactions on Automatic Control*, vol. 54, no. 1, pp. 4861, 2009.

[35] A. Nedíc, A. Ozdaglar and P. A. Parrilo, “Constrained Consensus and Optimization in Multi-Agent Networks,” in *IEEE Transactions on Automatic Control*, vol. 55, no. 4, pp. 922-938, April 2010.

[36] D. Niculescu and B. Nath, “Ad hoc positioning system (APS),” *Global Telecommunications Conference (GLOBECOM)*, vol. 5, pp. 2926-2931, 2001.

[37] N. Patwari, J. N. Ash, S. Kyperountas, A. O. Hero, R. L. Moses and N. S. Correal, “Locating the nodes: cooperative localization in wireless sensor networks,” in *IEEE Signal Processing Magazine*, vol. 22, no. 4, pp. 54-69, July 2005.

[38] N. Patwari, A. O. Hero, M. Perkins, N. S. Correal and R. J. O'Dea, "Relative location estimation in wireless sensor networks," in *IEEE Transactions on Signal Processing*, vol. 51, no. 8, pp. 2137-2148, Aug. 2003.

[39] M. Rabbat and R. Nowak, "Distributed optimization in sensor networks," *Third International Symposium on Information Processing in Sensor Networks, 2004. IPSN 2004*, pp. 20-27, 2004.

[40] J. B. Rawlings, D. Q. Mayne, and M. M Diehl, *Model Predictive Control: Theory, Computation and Design*, 2nd edn., Nob Hill Publishing, 2017.

[41] A. Savvides, H. Park, and M, B. Srivastava. "The bits and flops of the n-hop multilateration primitive for node localization problems." *Proceedings of the 1st ACM international workshop on Wireless sensor networks and applications.* pp. 112-121, 2002.

[42] Q. Shi, C. He, H. Chen and L. Jiang, "Distributed Wireless Sensor Network Localization Via Sequential Greedy Optimization Algorithm," in *IEEE Transactions on Signal Processing*, vol. 58, no. 6, pp. 3328-3340, June 2010.

[43] B. T. Stewart, A. N. Venkat, J. B. Rawlings, S. J. Wright, G. Pannocchia, "Cooperative distributed model predictive control," *Systems & Control Letters*, Volume 59, pp. 460-469, 2010.

[44] T. S. Tay, W. Whiteley, "Generating isostatic frameworks", *Structural Topol.* no. 11, pp. 2169, 1985.

[45] F. Thomas and L. Ros, "Revisiting trilateration for robot localization," in *IEEE Transactions on Robotics*, vol. 21, no. 1, pp. 93-101, Feb. 2005.

[46] J. N. Tsitsiklis, "Problems in Decentralized Decision Making and Computation, Ph.D. dissertation", Dept. Elect. Eng. Comp. Sci., Massachusetts Institute of Technology, Cambridge, 1984.

[47] A. N. Venkat, I. A. Hiskens, J. B. Rawlings and S. J. Wright, "Distributed MPC Strategies With Application to Power System Automatic Generation Control," in *IEEE Transactions on Control Systems Technology*, vol. 16, no. 6, pp. 1192-1206, Nov. 2008.

[48] M. Zhu and S. Martinez, "On Distributed Convex Optimization Under Inequality and Equality Constraints," in *IEEE Transactions on Automatic Control*, vol. 57, no. 1, pp. 151-164, Jan. 2012.